

A LIBRARY OF SOFTWARE PROGRAMS FOR THE **IBM** PERSONAL COMPUTER

PC DISK

M A G A Z I N E

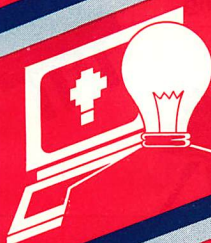
VOL. 1 NO. **4**

\$29.95

**USER
MANUAL**

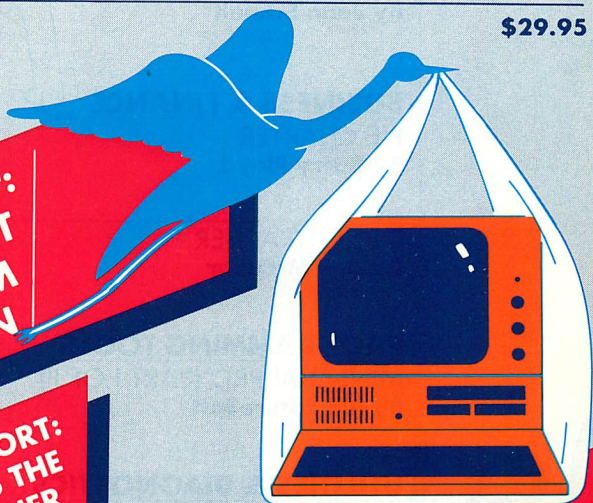
**SIZING UP PCjr:
A FIRST
LOOK FROM
PETER NORTON**

**TRANSPARENT SORT:
TEACHES THE
SHELL-METZNER
SORT BY
DEMONSTRATION**



**MINI-QUIKPRO—PART I:
A PROGRAM
THAT WRITES
BASIC PROGRAMS
FOR YOU**

**AIR TRAFFIC
CONTROLLER:
EXPERIENCE
CONTROL TOWER
ACTION**





ONTENTS

HOME/PERSONAL

SIZING UP PC *jr*

By Peter Norton

7

TRANSPARENT SORT

By John Schnell

11

BUSINESS & FINANCE

PIE CHARTER

By Barry Floyd

17

BIBLIOGRAPHER

By Don McCullar

21

PROGRAMMING TOOLS

MINI-QUIKPRO (PART I OF II)

By ICR FutureSoft

25

UTILITIES & DIAGNOSTICS

SORT/MERGE II

By Louis J. Cutrona, Jr.

33

GAMES

AIR TRAFFIC CONTROLLER

By Will Fastie and William Appelbaum

Original Design by David Mannering

39

ECONOMIC PIE

By John J. Springfield

45

DATA FILES

1983 FEDERAL TAX RATES

By Morris Effron

51

DEPARTMENTS

FROM THE EDITOR	5
-----------------	----------

SUBMISSION PROCEDURE	6
----------------------	----------

PROBLEM RECOVERY	50
------------------	-----------

SUBSCRIPTION OFFER	54
--------------------	-----------

COMING UP	55
-----------	-----------

TERMS AND CONDITIONS

LIMITED WARRANTY

In the unlikely event that the disk contained in this issue is defective in manufacture, return it within 30 days after purchase or receipt to PC Disk Magazine, 39 East Hanover Avenue, Morris Plains, NJ 07950, and it will be replaced at no charge. Merchandise for exchange must be accompanied by proof of purchase or a subscription label.

The publisher shall have no liability or responsibility to any purchaser or subscriber or to any other person or entity with respect to any liability, loss or damage directly or indirectly arising out of the use of the disk and the programs contained in this issue, including, but not limited to, any loss of business or any other incidental or consequential damages.

COPYRIGHT

Programs and Documentation Copyright © 1983 Ziff-Davis Publishing Company. All rights of reproduction in all forms and media strictly reserved.

The original purchaser/subscriber only is authorized to make copies solely for use in his/her own computer. Distribution, transmission, or transfer of copies to any other person is an infringement of copyright.

PC DISK MAGAZINE

PUBLISHER

André M.J. Van Hattum

EDITORIAL

Editor: Morris Efron

Associate Editor: Bohdan Shmorhay

Editorial Assistant: Laura Ninger

Assistant Editors: Matt Dolph, Alfred Poor, David Teisler,
David Wolff

ART & DESIGN

Cristina Botta, Robert Conrad, Don Menell Directions

CIRCULATION

Subscription Director: Chet Klimuszko

New Business Manager: Eric A. Bernhard

Renewal and Billing Manager: Shane Boel

Retail Sales Manager: Gerald Grossman

CONSUMER COMPUTER AND ELECTRONICS MAGAZINE DIVISION

President: Larry Sporn

Vice President, Marketing: J. Scott Briggs

Vice President, Circulation: Carole Mandel

Vice President, General Manager: Eileen G. Markowitz

Creative Director: Peter J. Blank

Editorial Director: Jonathan D. Lazarus

Promotion Manager: Ronni Sonnenberg

ZIFF-DAVIS PUBLISHING COMPANY

President: Richard P. Friese *President, Consumer Magazine Division:* Albert S. Traina
Executive Vice President, Marketing and Circulation: Paul Chook *Senior Vice Presidents:* Philip T. Heffeman, Sidney Holtz, Edward D. Muhlfeld, Philip Sine *Vice Presidents:* Baird Davis, George Morrissey *Treasurer:* Selwyn Taubman *Secretary:* Bertram A. Abrams

PC Disk Magazine is published 9 times per year, by Ziff-Davis Publishing Company, One Park Avenue, New York, N.Y. 10016. Subscriptions are available at \$179.00 for 6 issues. POSTMASTER: send address changes to *PC Disk Magazine*, 39 Hanover Avenue, Morris Plains, NJ 07950.

Editorial and Business Office: One Park Ave., New York, NY 10016. Editorial (212) 725-7770. For subscription inquiries and service write to *PC Disk Magazine*, 39 East Hanover Ave., Morris Plains, NJ 07950.

PC Disk Magazine is an independent journal, not affiliated in any way with International Business Machines Corporation. IBM is a registered trademark of International Business Machines Corp. Entire contents copyright © 1983 Ziff-Davis Publishing Company. All rights reserved; reproduction in whole or in part without permission is prohibited. ISBN # 087165-104-1 PC DISK MAGAZINE VOLUME 1 NUMBER 4



FROM THE EDITOR

After much ado and a protracted period of anxious anticipation, IBM's new home computer is finally here. Although it may be many months before you can actually get a PCjr, its significance and interest to the IBM PC community is immediate. We at *PC Disk Magazine* are pleased to offer our readers a special introduction to this new machine which combines the microcomputer expertise of Peter Norton with the unique character of our medium. The result, *Sizing Up PCjr*, is a program that enables you to explore the key features of this new home computer and compare them with the computer you currently use. We think that the opportunity to examine these capabilities online, rather than in print or pictures, provides an especially informative and exciting introduction.

As stated last issue, one of the most important and effective applications of software is in education. In addition to *Sizing Up PCjr*, our instructive software in this issue includes a program which demonstrates the operation of a powerful sorting technique. *The Transparent Sort* illustrates and explains the logic of Shell/Metzner sorting algorithm with an annotated, step-by-step demonstration. The computer's ability to teach, both about itself and other topics of interest, is a facility we will continue to emphasize in future issues of *PC Disk Magazine*.

One of the most interesting frontiers in software development is programs that write programs for you. Many people feel that with the proper interface such software could eliminate the need for any programming knowledge to develop new applications. Although you can't speak to it in English (the elusive perfect interface) *Mini-Quikpro* in this issue is an excellent example of such a program. This first installment creates your data management program. Next issue will provide the report program generator.

The other programs in this issue are no less valuable for my lack of space to describe them in detail. *Pie Charter* and *Bibliographer* perform useful business functions, while *Sort/Merge II* is a fast utility with wide applicability. No issue would be complete without some challenging diversion, which is what you can expect from *Economic Pie* and *Air Traffic Controller*.

As you continue to build a library of varied, quality software with *PC Disk Magazine*, you may think why not make a contribution, and some money, yourself? Why not indeed! Review the Submission Procedure in this manual and let us hear from you. Help us "push out the envelope" of the software frontier.



SUBMISSION PROCEDURE

What sets *PC Disk Magazine* apart from most other publications is that we want and need your direct involvement. That's why we developed the "Software Submission Plan." If you're interested in submitting software you've developed either on your own or with others, let us know, and we'll send you the Submission Plan booklet. We would like to give you an outline of our Submission Plan here in order to stimulate your imagination and your interest.

The Software Submission Plan provides an opportunity for software authors to profit directly from their work. Under the plan, *PC Disk Magazine* pays every published author a royalty on every issue sold which contains a copy of his or her software. Best of all, this opportunity comes without elaborate restrictions. Our desire is to license software for publication only for the disk magazine format.

To explore the considerable opportunities for publication in *PC Disk Magazine*, write to us and ask for a Software Submission Plan booklet. The address is:

PC Disk Magazine
Dept. 732
One Park Avenue
New York, NY 10016

Once you've received the Submission Plan, read it over carefully. If you feel your software fits the requirements set out in the plan, complete the enclosed Software Submission Agreement and return it to us. No program code or documentation should be sent along with the Submission Agreement.

Following receipt of the Submission Agreement, the editors of *PC Disk Magazine* will evaluate the submission. At that point, we'll either indicate a lack of interest in the software or, if we are interested, we'll issue a submission authorization number and ask you to provide us with an executable copy of the program along with documentation on 5 $\frac{1}{4}$ " diskettes. Again, no material should be sent until you receive a submission authorization number.

If we subsequently decide that we would like to publish the software in *PC Disk Magazine*, we will offer a Software Contract, which will include such items as royalties, advances, and program and documentation changes required (if any). You will be asked to complete and test any program modifications agreed to in the Software Contract, and *PC Disk Magazine* will conduct a formal validation of the program and documentation.

We estimate that this process, from our initial evaluation to publication, takes approximately 3 to 6 months. This estimate is dependent upon a number of factors, and the process may take more or less time for your submission.

We look forward to hearing from you.



SIZING UP PCjr

By Peter Norton

Special Requirements: None
Files Used: JUNIOR.BAS

The most anxiously awaited personal computer in memory (permanent or volatile) is finally here: the IBM PCjr. The IBM PCjr provides much of the capability of the original IBM PC at a relatively modest price, and it provides a host of new features and capabilities as well. By way of introduction to this new machine, we have combined the unique format of PC Disk Magazine with the special expertise of Peter Norton to bring you SIZING UP PCjr.

SIZING UP PCjr is a program designed to help our readers learn about some of the most exciting new features of the PCjr, whether they are using a PCjr or not.

SIZING UP PCjr is an interactive program, written in IBM BASIC, which explains and demonstrates many PCjr features. The program will showcase four feature categories—sound, video modes,

speed and cartridges—with three demonstrations in each category. These twelve parts are all selected from easy-to-use menus.

You don't have to have a PCjr to size one up—SIZING UP PCjr will work fine on any PC or PC-XT, Compaq or other IBM-compatible computer. Running on a PCjr, however, will show off the full capabilities of this program. Running it on another computer will do two things: teach you about the PCjr, and show you how the capabilities of your computer stack up against those of the new PCjr.

This program uses color extensively, so it is most interesting with a color display. However, it will work quite well with either a monochrome setup or a black and white monitor on the color/graphics adapter.

SIZING UP PCjr was written to help you learn as much as possible about the new PCjr—not only through the program's operation, but also by the way that it was written. If you like exploring BASIC programs, you are invited to list out the program code and learn the techniques that were used to demonstrate the PCjr's features.

START-UP

To use the program, you must first load BASIC into your computer. Put a diskette with the file BASIC.COM in the default drive and type:

BASIC 

Then put the *PC Disk Magazine* diskette into your default drive and type:

RUN“JUNIOR 

You will be asked if you are using a color monitor. Just type in “Y” or “N” as appropriate, followed by the Enter key.

EXPLORING JUNIOR

The first screen to appear after start-up will be the program's main menu, which consists of five categories:

- 1. Sound Effects**
- 2. Video Modes**
- 3. Speed Trials**
- 4. Software Cartridges**
- 5. Return to DOS**

The “Sound Effects” portion of this program will demonstrate the sounds that the PCjr can produce. “Video Modes” will explain and demonstrate ten different ways to display information on the video screen. “Speed Trials” will compare the execution speed of different operations on the PCjr (or whatever computer you are using), against the IBM PC. “Software Cartridges” will explain how cartridges work, and also seek out and report cartridge-related software on your system. Each of these four topical choices leads to another menu, with a choice of three items to explore concerning the main topic. Press:

5

on the main menu to exit from the program and return to DOS.

SOUND EFFECTS

By pressing the "1" key from the main menu you will see the following submenu:

1. Sound Paths
2. Multiple Voices
3. Noise Demonstration
4. Return to Main Menu

If you select "Sound Paths" you will receive an explanation and demonstration of how the PCjr can use both internal and external speakers for the sound it creates. "Multiple Voice" will demonstrate the PCjr's ability to generate more than one sound at a time. "Noise Generation" demonstrates the PCjr's special ability to create noises which can be used in games. Option 4 will return you to the main menu. Note that after each selection, you will be returned to this submenu for another sound topic. You must choose option 4 to return to the main menu.

VIDEO MODES

Option 2 of the main menu calls up the following "Video Mode" menu:

1. Video Information
2. List of Video Modes
3. Demonstration of Modes
4. Return to Main Menu

Option 1 explains the general principles of the IBM video modes and their applications to the new PCjr.

Option 2 lists all the video modes, so that you can compare the features of the modes, and see which ones apply to the PCjr.

Option 3 demonstrates the video modes, showing what will and won't work on your computer, and explains the BASIC statements used to switch to each mode.

Option 4 returns you to the main menu.

SPEED TRIALS

"Speed Trials" offers three different benchmarks with which to compare the speed of your computer against the IBM PC. Each of these benchmarks will perform a simple set of operations, running for about 10 seconds, and report how long they took, compared to a sample run done on an ordinary IBM PC. The purpose of these tests is to demonstrate the PCjr's relative speed, which is somewhat slower than the PC. However, these same tests can be used to test the speed of whatever computer you are using, including the faster speed of a hard disk system.

The first test is called "Pure Computing Speed." In this comparison, two large equations are calculated forty times, and the results from your machine are matched with the results from the standard PC. In the second trial, you can test the time it takes both computers to generate a video display thirty times. The third test is used to determine how quickly the computer can access its disk drive and create, write to, and erase a small file named PCJUNIOR.TST. The diskette used by *PC Disk Magazine* is write-protected, so in order to run this third test you will have to use your own diskette.

While these speed trials will indicate relative performance, you should take their results with a grain of salt. In developing these tests, a surprising degree of speed difference was discovered

**YOU
GET
THREE
CHOICES
OF SPEED
TRIALS**

2.1 Testing Pure Computing Speed

For this test we repeatedly

```
Calculate ATN(TAN(SQR(.5)^2.0))  
Calculate LOG(2.71829^.5)
```

On an ordinary PC this took

10.3 seconds, for 40 repetitions

On this computer it takes

11.0 seconds, for 40 repetitions

7 % longer

Press any key to continue ...

The speed benchmarks

among ordinary PCs, and Compaqs as well. The results will be influenced by the version of DOS, the version of BASIC, the type of display screen, and especially by the disk or diskette used. Even the format of a diskette, and how full it is, will dramatically influence the disk speed test. In fact, you might want to use option 3, the disk speed test, to explore these factors.

SOFTWARE CARTRIDGES

This section of the program is designed specifically for the PCjr, which is equipped with two software cartridge ports. The cartridges are Read Only Memory (ROM) and are used for games and programs. They are identified by a signature code and a list of their contents.

Option 1, "Information on Cartridges," will give a short explanation of how the cartridges work.

Option 2 will test to see if either of the PCjr's cartridge slots are in use, and then report what it can discover about the cartridges, including a list of any commands which can be executed on the cartridges.

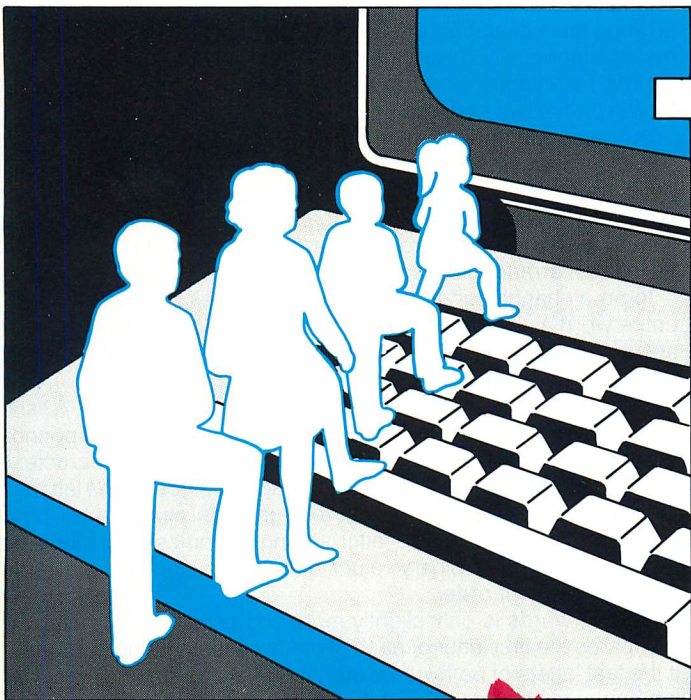
Option 3 will search the computer's memory for any software with a signature mark that is similar to that used by the cartridges. For example, on a PC-XT, or an IBM PC or Compaq computer equipped with an IBM-style fixed disk, this program will detect and report the fixed disk support program.

EXITING

Whenever you wish, you may exit the program by returning to the main menu and pressing:

5

This will return you to DOS.



TRANSPARENT SORT

By John Schnell

Special Requirements: None
Files Used: TRANSPARENT.BAS

"Sorting" is a programming technique that organizes and stores information in alphabetical or numerical order. It frees you from sequencing information manually before saving it, and becomes invaluable when you need to order hundreds or thousands of items. TRANSPARENT SORT is a tutorial sorting program which allows you to watch the sorting process in action. By watching a sort work line by line, you will be better able to understand the programming technique involved.

BACKGROUND

Suppose you have a list of items to be sorted. A sorting subroutine is a step-by-step procedure that the computer follows in order to sort the items. During the sorting process, the computer compares

**THIS
PROCEDURE
CONTINUES
UNTIL ALL
ITEMS
HAVE BEEN
COMPARED,
AND ALL
EXCHANGES
MADE**

two items at a time. If it determines that one item should precede the other, the items exchange places on the list. This procedure continues until all the items have been compared to each other, and all the exchanges have been made. Ultimately, all the items on the list are placed in sequential order.

There are two kinds of sorts: alphanumeric and numeric. An alphanumeric sort will take any array of letters, words, or names and arrange them in alphabetical order. A numeric sort will take any array of numbers and arrange them from lowest to highest. Both sorting techniques work on the principle of compare and swap. However, the two sorts approach the task differently.

In a numeric sort, the computer evaluates each number to be sorted as a single, large, binary expression. For example, the computer reads the number 48 as the binary expression 00110000. When two expressions are compared, the computer places the item of lower value before the one of higher value until the items have been arranged in ascending numerical order.

In an alphanumeric sort, the computer does not evaluate each entire word as a single, large, binary expression. Instead, it evaluates each letter in a word individually. Each character of the alphabet corresponds to a binary number, called its ASCII value. (ASCII stands for American Standard for Information Interchange. A list of these character-to-number correspondences appears in Appendix G of the IBM BASIC manual.) The ASCII value for each character is itself a one-byte expression. Since the ASCII values for the letters of the alphabet range from lowest to highest in alphabetical order (e.g., A is 65 and Z is 90), letter sorting becomes very much like number sorting. The binary values of the characters are compared to determine their order.

Sorting words is only slightly more complicated. The computer compares the first letter of each of the two words being compared. If the first letters of both words are the same, the second letters will be compared to determine if the words should be swapped. If the second letters are the same the computer compares the third letters, and so on.

START-UP

To see one way in which a computer actually performs this operation, using a method known as the Shell/Metzner sort, load Advanced BASIC into your PC by typing:

BASICA 

Then place your *PC Disk Magazine* diskette into your default drive and type:

RUN"TRANSORT 

THE MAIN MENU

The main menu will present four options: Transparent Sort, Non-Transparent Sort, Sort Listing, and End Program. To choose an option, press:



or



to move the inverse video box to the option you desire. When your choice is highlighted, press:

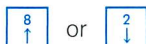


to make your selection.

Start by choosing the TRANSPARENT SORT.

THE TRANSPARENT SORT

The TRANSPARENT SORT is a step-by-step visual demonstration of the Shell/Metzner sort. If you select this option, you will reach a second menu. You must choose some initial arrangement for the 26 letters which you will see being sorted into ascending alphabetical order. Each initial arrangement will give varying final sorting statistics depending on how scrambled the letters are to begin with. To select an option, move the inverse video line up or down by pressing:

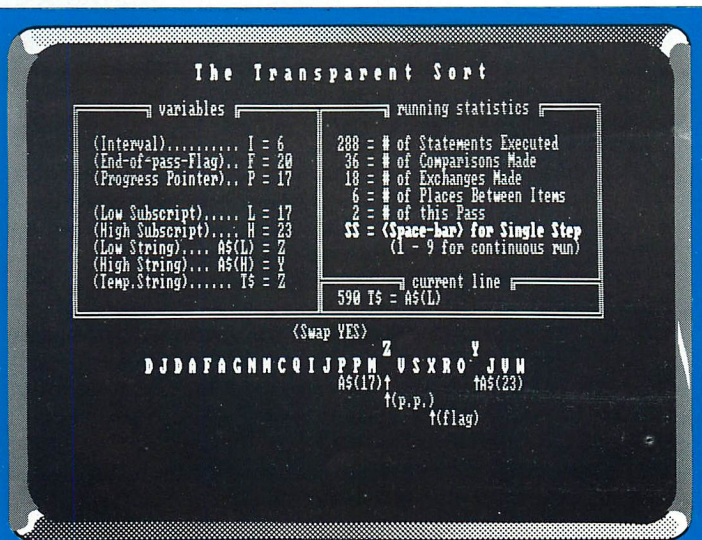


When the desired option is highlighted, press:



to make your choice.

**YOU
MUST
CHOOSE
AN
INITIAL
ORDER**



The Transparent Sort in progress

Note that you can create your own starting order for the letters to be sorted by using the "Select Your Own Alphabet" option and following the on-screen prompts. When you have decided on the initial arrangement of the letters, the main display will appear. It is on this screen that you can watch the sort execute.

Press any key to start the demonstration. Once it has begun, you may control the speed of the demonstration by pressing keys 1 (fastest) to 9 (slowest). The default speed is set at 5. You may also single-step through the program, one statement at a time, by press-

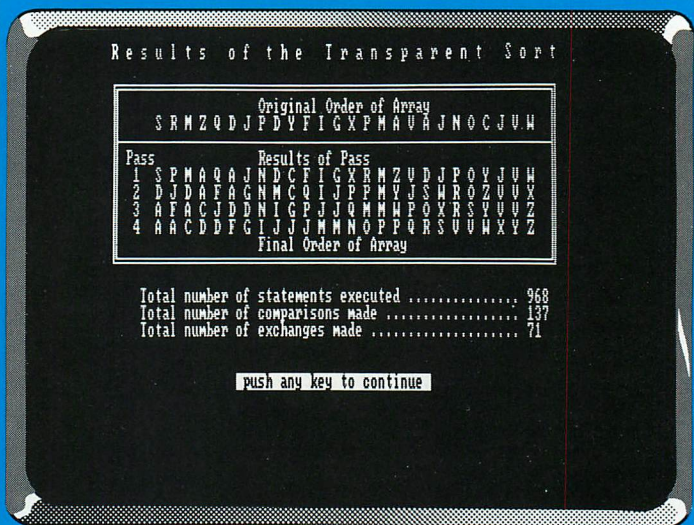
ing the Space Bar. To restart continuous running, press any numeric key. In order to cancel the sort and return to the main menu, press:

M

During a sorting run, the 26 letters to be sorted are displayed across the center of the screen. The letters will switch positions as dictated by the BASIC line statements being executed. The BASIC program line being executed appears in the "current line" box above the alphabet, while an associated explanatory message appears at the bottom of the screen.

Arrows moving below the alphabet indicate the items being compared and the position of the Flag and the Program Pointer.

As you watch the sorting process, you will better understand the principle behind the Shell/Metzner sort. Generally, the program makes 4 passes on the letters, ordering them more accurately each time. On the first pass it compares each letter with the letter 13 places after it, and swaps them if they are out of order. On the second pass it will compare letters 6 places apart, on the third pass, 3 places apart, and on the final pass, 1 place apart.



A results screen shows a summary of the sorting process.

If, beginning with the second pass, the program switches two letters, it will also compare the lower swapped letter with the letter 6 places earlier, and exchange them too if necessary. By repeating this process at each level, the computer reaches the final order.

The box at the top of the main screen displays the status of each variable, some pertinent statistics about the level of the program, and the current line of the BASIC program being executed. It updates these items after each program line.

At the end of a sorting run, another display will show the original array of 26 letters to be sorted, the arrangement after each pass, the final sorted array, and the final statistical totals for the sort.

NON-TRANSPARENT SORT OPTION

The Non-Transparent Sort demonstrates the true sorting speed of the Shell/Metzner sort. (The fastest setting in the TRANSPARENT SORT mode is many times slower than an actual sort.) If you select this option, you will again see a second menu consisting of alternative arrangements for the letters being sorted. You may choose any alphabet arrangement you selected under the Transparent Sort mode, and witness the true speed of this algorithm.

THE SORT LISTING

The Sort Listing option displays the following BASIC subroutine:

```
510 I=26
520 I=INT(I/2)
530 IF I=0 THEN RETURN
540 F=26-I
550 P=1
560 L=P
570 H=L+I
580 IF A$(L) <= A$(H) THEN 640
590 T$=A$(L)
600 A$(L)=A$(H)
610 A$(H)=T$
620 L=L-I
630 IF L > 1 THEN 570
640 P=P+1
650 IF P > F THEN 520
660 GOTO 560
```

This sorting method, or "algorithm," is called a Shell/Metzner sort, after its originator, D.F. Shell, and its improver, M. Metzner. It is an efficient sort, used in a variety of applications. This particular demonstrative application will alphabetize 26 letters. If this listing isn't clear, don't worry. You can easily make it so by returning to the main menu and choosing the TRANSPARENT SORT option. For those who do want to follow the code at this point, the subroutine can be clarified as follows:

Line 510 describes the value of I as equal to the total number of items to be sorted (in our case 26).

Line 520 establishes the interval (number of spaces) between the two items to be compared.

Line 530 ends the sorting process when the interval between items reaches zero.

Line 540 sets the position of the end-of-pass Flag.

Line 550 sets the position of the Progress Pointer.

Line 560 sets the low subscript (L) equal to the value of the Progress Pointer.

Line 570 establishes the number of places between low item and high item to be compared.

Line 580 compares two items in the list A\$. If no swap is made, then the program jumps to line 640. If a swap is to be made, the program goes on to line 590. Lines 590-610 perform the swap.



**HIS
DEMO WILL
ALPHABETIZE
26 LETTERS**

**YOU
CAN
RETURN
TO THE
MAIN
MENU
AT
ANY
TIME**

Line 590 stores the value of the low item in temporary storage (T\$) so it will not be lost.

Line 600 copies the contents of the high item storage into the low item storage.

Line 610 moves the contents of the temporary storage into the high item storage. The swap is now complete.

Line 620 reduces the value of the low subscript by the amount stored in variable I.

Line 630 asks if the low item is equal to or greater than one. If the low item is equal to or greater than one, the program jumps back to line 570 where a comparison between the item just moved and another lower item can be made, with the possibility that it will be moved even farther down the list.

Line 640 moves the Progress Pointer to the next higher item.

Line 650 asks if the value of the Progress Pointer is greater than that of the Flag. If it is, we have reached the end of this particular pass and the program jumps to line 520.

Line 660 routes the program back to line 560 where the process starts on two new higher items.

EXITING

You can stop the program while in the TRANSPARENT SORT mode by pressing:

M

This will return you to the main menu. You can then select the "End of Program" option to return to Advanced BASIC.

BACK ISSUES OF PC DISK MAGAZINE

Add to your *PC Disk Magazine* collection today, while copies are still available. The more complete your library of back issues, the more enjoyment and use you can get from your personal computer. The earliest available back issue is Volume 1, Number 1. Order any issue you want, being sure to specify volume and issue number, and quantity, for each. This offer is subject to availability. If a particular issue is out of stock, your payment for that issue will be refunded promptly.

Back issues of *PC Disk Magazine* are priced at \$31.95 each, which includes postage and handling.

Please write to:

PC Disk Magazine

CN 1916

Morristown, N.J. 07960

Be sure to include volume and issue number, as well as the number of copies desired, and the address to which you would like those copies sent. Please enclose your payment with your order.



PIE CHARTER

By Barry Floyd

Special Requirements: Color/Graphics Adapter
Graphics Printer for Hardcopy

Files Used: PIECHART.BAS
PCDUMP.COM

Are your business facts and figures becoming hard to swallow? Do all those numbers stick in your throat or form an indigestible lump in your boss's stomach? Don't reach for the Alka-Seltzer! Let PIE CHARTER take all that data and shape it into a series of beautiful, digestible pie charts. The relationships and proportions become clearer and decisions become more obvious. You can recast confusing data as a computer-generated treat.

PIE CHARTER allows you to label up to ten pieces of numerical data and combine them into a pie chart. You can create up to four different charts on one screen and print the results on your graphics printer.

START-UP

Before running the *PIE CHARTER* itself, you need to load a subroutine that enables the computer to dump a copy of your pie chart from the screen to your printer. While in DOS, put the *PC Disk Magazine* diskette in the default drive and then load this subroutine by typing:

PCDUMP 

The subroutine will remain in memory until you turn your computer off, so you need load it only once during a work session. To start *PIE CHARTER*, load Advanced BASIC into your PC from a system disk by typing:

BASICA 

then put the *PC Disk Magazine* diskette into your default drive and type:

RUN"PIECHART 

The main screen that now appears lists all existing pie chart files and prompts you to enter a filename. Note that the program accepts only the filename; the file extension (.GRF) is added automatically. Filenames are limited to eight characters. Entering a filename not listed as an existing graph file will create a new file with that name and a .GRF extension. Entering an existing filename will call up that file.

FUNCTION KEYS

The function keys have specific, consistent purposes throughout the program. Each screen in the program will list the currently active keys and explain their functions. The only exception to this is the pie chart screen, which for aesthetic reasons does not contain a function key list.

GRAPH SETUP

Once you specify a graph file to work on, pressing:



F5 

will allow you to set up the labels for the graph. When the setup screen appears, enter the graph labels that you want to have appear on your final pie chart display.

When a chart is displayed, the slices of each pie are labeled with small symbols which are keyed to the data labels you assign each data point. If you would like these names and symbols displayed on the pie chart, simply set the value of the "Display All Text (Y/N)" parameter on the setup screen to "Y." Setting this parameter to "N" suppresses display of titles and the label legend.

The setup screen also allows you to label each of up to four pies (representing four data sets) displayed on the graph screen.

The procedure to set or alter any of these parameters is first to use the up and down arrow keys:

 or 

to move the cursor to the appropriate line. Then to open that item for editing, press:



Single quote marks will appear to indicate that the item is awaiting your entry. Type in the information you want, then press:



and the program will accept the new value. This is the standard procedure for entering and editing all data and labels in *PIE CHARTER*. Once you have set up the graph parameters, you may proceed to the Edit Data Screen by pressing:



DATA ENTRY/EDIT

A separate data entry screen exists for each data set (pie) within a given graph file, but all have an identical format. On this screen you enter a data value and a label for each slice of the pie, up to ten slices in all. Enter data by pressing:



Quote marks will appear in the first line of the data column. Enter the numerical data according to the standard procedure.

Labels may be added or existing data altered in a manner similar to the Setup Graph procedures. First use all four cursor movement keys to position the cursor on the appropriate item (data or label), then press:



to open the item for editing. Enter your new value and press:



to conclude the process. The F2 key will delete the entire line where the cursor is currently positioned, across all data sets.

To enter data for another pie in the same file (another "set"), press:



This will move to a new Edit Data Screen. The labels from the first set will appear in the label column for the second set. This makes it easy to draw several related pies with different sets of data. You can alter these data sets, delete them, or graph them, according to your needs.

GRAPHING

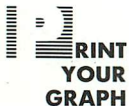
Once all of your data has been entered and your labels set, pressing:



will draw a pie chart with one pie for each set of data. The program does this by adding all the data values for each set and assigning a proportionally sized slice of the pie to each entry.

The slices are labeled with symbols and the key is given on the left side of the screen (provided you have not toggled it off). Note

A stylized pie chart icon consisting of a circle with a vertical line through the center and a horizontal line at the bottom, creating a shape like a 'C' with a dot in the middle.
**DATA
ENTRY
SCREEN
EXISTS
FOR
EACH
DATA
SET**



that the directory of function keys does not appear at the bottom of this screen. The applicable function keys and their actions are:

F5

returns you to the Setup Graph Screen.

F6

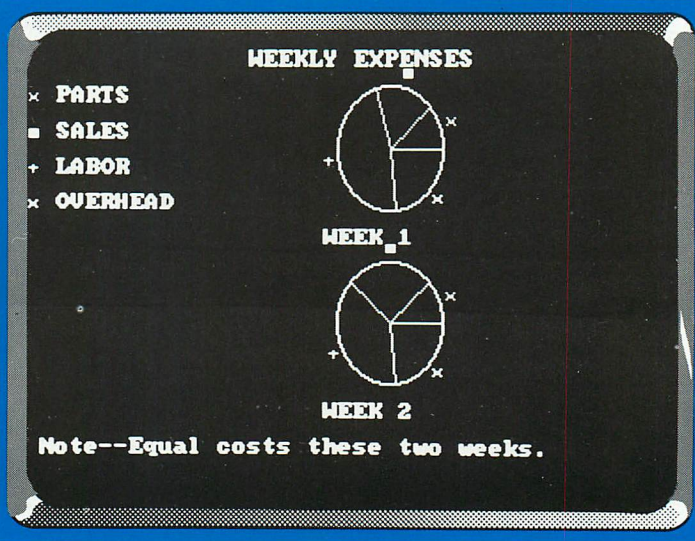
returns you to Edit Data Mode.

F1

will make a printed copy of your graph on any IBM-like printer with the graphics feature.

F9

will make a printed copy of your graph on any Okidata printer.



A sample pie chart of weekly expenses

Both print functions will leave you with the graph screen still displayed, so you must then select one of the other two active function keys to proceed.

EXITING

Any time that the F10 key appears in the function key legend at the bottom of the screen, you can use it to exit the program and return to Advanced BASIC.



BUSINESS & FINANCE



BIBLIOGRAPHER

By Don McCullar

Special Requirements: None

Files Used: BIBLIO.BAS
REFS.DAT

Your office resembles San Francisco after the earthquake; clip-pings, magazines and books are stacked and scattered across the deskscape like so much rubble. The trouble is that you're an avid and conscientious reader; whenever you find something of interest, you save it for later reference. Now you must spend hours searching through the piles trying to locate that article you had so diligently saved two years ago.

With BIBLIOGRAPHER, that reference can once again be at your fingertips, because you can quickly scan all the potentially relevant entries at the touch of a couple of keys. BIBLIOGRAPHER lists references by topic for easy access, and even has enough space within the individual entries to allow brief descriptions of each. This capability assists you in deciding whether a particular reference is the

one you want. You can update your files at any point by making your topic listings more detailed or precise.

For the individual who wants to organize his library, or the business person who requires constant and immediate access to reference material, *BIBLIOGRAPHER* makes an excellent librarian.

BACKGROUND

All your references are stored in a single file, REFS.DAT, which the program creates and maintains for you. You enter a reference by supplying seven items of information. These items refer to specific aspects of a reference; for instance, the "subject" identifies the general topic; the "source" indicates the medium (magazine, book, or tape); "other information" provides a one-line abstract of the reference. The other four fields—date, title, author, and page number—are all self-explanatory. Your only restriction is the number of characters each item will accept. You have the ability to alter any item of information for any reference on file, as well as delete a reference entirely.

The program lets you list all the references in your file, in either an abbreviated form on the screen or with all information recorded on the printer. You may also selectively retrieve a group of references for review, but only by subject. No field other than subject can be the basis of a research request.

START-UP

To start *BIBLIOGRAPHER*, you must first load Advanced BASIC into your PC. While in DOS, put a diskette with the file BASICA.COM in your default drive and type:

BASICA 

Then place your *PC Disk Magazine* diskette into your default drive and type:

RUN"BIBLIO 

BIBLIOGRAPHER will offer you five options on its main menu. "Add New Reference(s)" allows you to create new reference entries. "Alter/Delete Existing Reference(s)" lets you change the particulars of existing references or delete them entirely. "Lookup By Subject" gives you access to references by topic, while "List All References" will give you a complete list of all entries and their corresponding reference numbers (an identification which you will need in order to make corrections). Either of the last two options offer a choice of the screen or your printer for the resulting list of references. "Quit" exits the program and returns you to Advanced BASIC.

A FEW FOOTNOTES

BIBLIOGRAPHER is a simple program to use. To choose an option from the main menu, you need only press the corresponding number. To enter all other information, you must first enter the data in response to *BIBLIOGRAPHER'S* prompts, then press:



When adding references, *BIBLIOGRAPHER* prompts you to enter information for each of the seven fields that constitute an individual

reference. Each field has a prescribed maximum (but no minimum) number of characters which can be entered into it; you may, if you choose, leave any category blank.

Every reference in your bibliographic file has a reference number associated with it. The computer assigns this reference number at the time you create the entry, and the number becomes uniquely and unalterably associated with that reference. The reference number has no particular significance—it is based entirely on the order of entries—but it gives the program a simple hook by which to quickly retrieve any reference entry in your file. When making corrections, you will be asked to supply the record number of the reference in question. All record numbers for all references in your file can be found by using option 4, "List all References," from the main menu.

At any point in the program, you may press:

A small icon of a keyboard key with the word "Esc" on it, enclosed in a blue square border.

to abort the operation in progress and return to the main menu. In a similar manner, when you are confronted with the first prompt that follows a main menu selection, you can press:



to return directly to the main menu itself.

Armed with this information, we can begin examining the individual options presented on the main menu.

CREATING REFERENCES

To create references, select option 1 from the main menu. Your screen will show an arrow followed by the prompt "Enter Subject". This topic field can hold a maximum of 10 characters and no matter how you enter them, all letters are stored as capital letters. The next field, the "Source," (that is, the medium or specific publication—book, TV, *New York Times*, *PC Disk*, etc.), can hold up to 16 characters. You will then be prompted to "Enter Date." This is an eight character field which can hold a date in the form of your choice (MM/DD/YY or Mon, YYYY) or a volume and issue number, or any appropriate time identifier. Following date, you are asked to "Enter Title" of the reference (maximum of 30 characters), "Enter Author" (maximum of 20 characters), "Enter Page Number" (maximum of 4 characters), and "Enter Other Information" which helps you to further identify the reference content (maximum of 40 characters). After entering a field (or leaving any of the fields, except Subject, blank) press:



to progress to the following field. If you leave Subject blank and press the Enter key, you return directly to the main menu.

Once all seven fields have been completed, your screen will present you with the beginning of a new reference record to fill in. If you have a second item you want to enter into the reference files, repeat the above steps. If not, press either the Escape key or the Enter key to return to the main menu.

**ALL
SUBJECTS
ARE
STORED AS
CAPITAL
LETTERS**

If in the course of making a reference entry you wish to cancel the entry in progress, or go back and change a previous field value, use the Escape key. Each time you press:

Esc

you will go back one field in the prompting sequence. To cancel an entry in progress, keep pressing the Escape key until you are back at the Subject prompt. Then press the Escape key one more time (or the Enter key) to return to the main menu.

Should you want to review the references you've entered, select option 4, "List All References," from the main menu. *BIBLIOGRAPHER* will ask if you want output on the screen or your printer, and then display all the references sorted by subject, and source within subject. On printed output, each reference will appear with all its information. On screen output, each reference appears with its number, subject, source, date and title. Use the Enter key to continue paging forward through the reference entries on screen display. Once you have completed your scan, press the Escape or Enter key to return to the main menu. You can cancel the printing process at any time by pressing the Escape key.

RETRIEVAL BY SUBJECT

There may be times when you're interested in researching a particular subject in *BIBLIOGRAPHER'S* reference lists. Select option 3 from the main menu, "Lookup By Subject," to view all references entered under a particular topic. If there are more entries in that topic than can be displayed on one screen, *BIBLIOGRAPHER* will prompt you to press the Enter key to continue reviewing the topic. Both screen and printer output can be interrupted at any time by pressing:

Esc

which will return you to the main menu.

MAKING CORRECTIONS

Suppose, while reviewing the references, you come across an error. Determine the appropriate record number and return to the main menu. To alter a reference, select option 2. The program will present the reference information and prompt you to enter one of eight letters. The first six letters correspond to reference information fields, while the last one is "N" to select another record for alteration. To correct one of the references, press the appropriate key. You are then asked to enter the correct information for the field chosen. Type in the new information. Once the program has accepted the change, it will again prompt you for the field you want to change, allowing you to make another alteration to the same record.

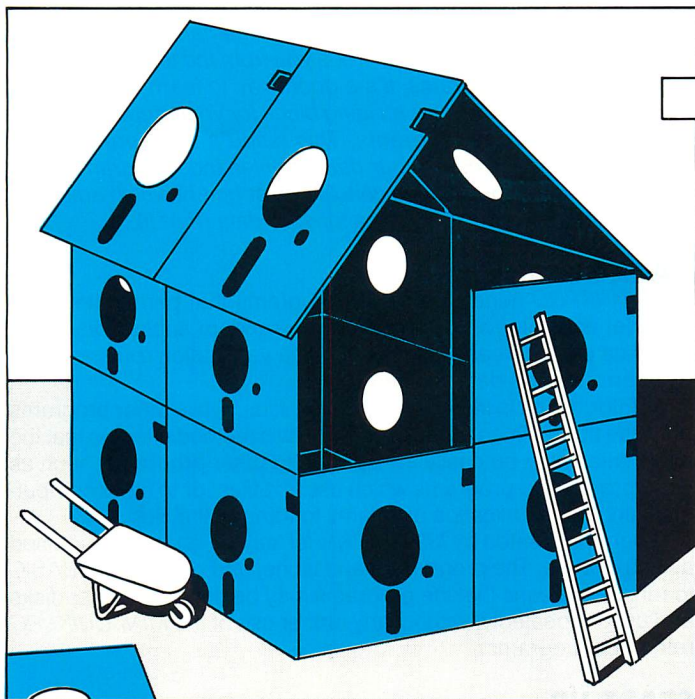
If you want to erase a reference, type in that reference's record number and press the Enter key, then type "E" for erase, then confirm your intention to delete this reference. If you have no more corrections to make, press the Escape key to return to the main menu.

When you have completed your work in *BIBLIOGRAPHER*, type:

5

on the main menu to exit the program and return to Advanced BASIC.

LOOK
IT UP
BY
SUBJECT



MINI-QUIKPRO

By ICR FutureSoft

Special Requirements: None

Files Used: QUIKPRO1

RANDWRT

RANDWRT2

Soon after you start using your IBM PC, you will probably want to save lists of information so that you can use the same information over again without having to re-enter it. Some aspects of data file management can be tricky, and it takes careful planning to write the many lines of program code required just to save and retrieve a simple set of names and addresses. MINI-QUIKPRO spares you that effort.

MINI-QUIKPRO is like a robot that writes BASIC programs for you. Simply tell it what information you want to keep, and it will automatically write a complete program to manage that information, ready for you to run with a single command. The key to this process

**THE
PROGRAMS
CREATED
BY MINI-
QUIKPRO
MAY BE
LISTED AND
MODIFIED
AS YOU
CHOOSE**

is the professional-quality data entry screen you design quickly and easily, in dialogue with MINI-QUIKPRO.

You can also list the programs created by MINI-QUIKPRO to see how they work. MINI-QUIKPRO uses plenty of REMark statements so you can see what each piece of code does. Armed with this information, you can easily go into the programs that MINI-QUIKPRO creates for you and make modifications if you choose.

MINI-QUIKPRO is more than a file management program; it is a BASIC program generator that frees you from the tedium of writing file management programs, it's a good way to learn about BASIC file programming, and it's a building block for your creation of high-quality applications programs. This issue's segment of MINI-QUIKPRO (Part I) creates your data maintenance program. In the next issue we will publish the concluding segment which automatically creates reporting programs for your data (Part II).

BACKGROUND

MINI-QUIKPRO generates BASIC programs that permit the entry, storage, and retrieval of information in random access disk files. The data entry screens incorporate data verification routines to reduce errors when data is entered.

All information is stored as ASCII text files. While other programs may use abbreviated forms of storage, this method ensures that the information may be easily transported to other programs, such as word processing programs which use text files, or to other computers using communication programs to transfer the files.

Programs created by MINI-QUIKPRO may be listed and modified as you choose. The programs stand alone, and only require BASIC to run. This means that the programs may be copied to other disks and used repeatedly, without any further use of the MINI-QUIKPRO reference programs.

START-UP

In order to use MINI-QUIKPRO, you must transfer the necessary programs from the PC Disk Magazine diskette to a formatted diskette with room for the additional files generated by the program. To do so, place your PC Disk Magazine diskette into the default drive and type:

COPY QUIKPRO*.* B: 

COPY RANDWRT*.* B: 

where "B:" is the drive holding the formatted work diskette.

To start MINI-QUIKPRO, load BASIC into your PC, specifying the proper record buffer size, by typing:

BASIC /S:255 

Record Length

The combined size of all your data fields, including the decimal points of numeric fields, cannot exceed 255 characters. Then place your MINI-QUIKPRO work diskette into your default drive and type:

RUN"QUIKPRO1 

The program will display the *QUIKPRO* menu, composed of two simple choices. Press:

1 

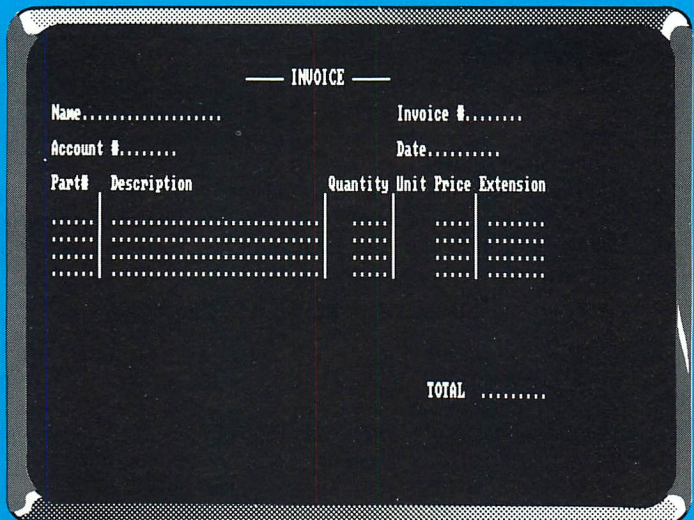
to begin using *MINI-QUIKPRO*, or press:

2 

to exit the program and return to DOS.

THE FORM DESIGN SCREEN

When you elect to run *MINI-QUIKPRO*, the next screen to appear is the Form Design Screen. It is labeled with numbers across the top and letters down the side. The prompt at the bottom of the screen will ask which screen line you wish to work on next, as designated by one of the vertical list of letters.



INVOICE				
Name.....		Invoice #.....		
Account #.....		Date.....		
Part#	Description	Quantity	Unit Price	Extension
.....
.....
.....
.....
TOTAL			

A sample Form Design Screen

You define the data and the data entry screen for your program by using this screen. Your entries here also determine the structure of the file used by your program to store the data. You define the data entry screen by drawing in the different "fields," that is, data items, that are to be filled with real values in much the same way that you might design the format for an index card in a file for names and addresses or other information.

You may make the screen easier to use by prefacing your data fields with labels and by providing other information on the screen. *MINI-QUIKPRO* not only lets you use letters and numbers in your layout, but also lets you draw boxes and other shapes with the IBM PC's special graphics characters. With a little practice, you can draw truly professional-looking screens.

Start by selecting a line near the top of the screen, entering its letter in response to the prompt at the bottom of the screen. The cursor will move up to that line on the screen; here you may want to type a

 REFACE
**YOUR
DATA
FIELDS
WITH
LABELS**

CREATE FIELDS TO HOLD DATA BY TYPING EQUAL SIGNS

title for the data entry screen. This could be "Club Member Names and Addresses," or whatever information you wish to store. Use the Space Bar to move the cursor towards the middle of the line so that your title is centered. When you have finished typing the line, press:



and the cursor will return to the bottom of the screen, waiting for you to choose the next line to draw on.

If you make a mistake while typing on a line, you may use the Backspace key to erase one character at a time until you have removed the mistake. Then retype the line the way you want. If you have already pressed the Enter key, and the cursor has returned to the prompt line, simply select that line again and retype it.

You may draw with graphics characters by pressing:



This will change the cursor from a solid block to a cross-hair. This key combination works as a "toggle switch;" each time you press it you will switch from text to graphics, or graphics to text. In the graphics mode, all the keys produce different symbols on the screen (see the Graphics Chart in your BASIC manual, Appendix G). Using these, you can draw a wide variety of boxes, special characters, and other details.

You create fields that will hold data by typing a string of equal signs "=". Be sure to put as many equal signs as will be required for the maximum number of characters that will go in that space. For example, you might have a line like the following:

Town: ===== State: == Zip: =====

There are 7 spaces for the "Town" field, two spaces for the "State," and five for the "Zip." Note also that you may put more than one field on a line.

Remember that if a field is to hold numeric information, a decimal point takes up one space. If you want to enter the number 123.45, you will need to have at least six spaces in that field.

Once you have designed your screen the way you want it, press the Enter key when asked for the line you want to work on. *MINI-QUIKPRO* will ask to make sure you are finished. If you answer "Y," you will then go on to the next step in defining your program. If you answer "N," you will be able to continue work on the Form Design Screen.

DEFINING THE DATA

First you must define which field will be the "key field" for your data file. This feature allows you to recall information from your file most rapidly. You will still be able to find records based on the other fields, but you should make the most important field the key field. In general, this should be a field that distinguishes one record from another, such as last name or social security number.

The data entry screen is displayed with each field numbered. The prompt on the bottom of the screen will ask "Which field will be the primary key?". Simply type the number of the field you wish to use

0	1	2	3	4	5	6	7
123456789012345678901234567890123456789012345678901234567890123456							
A							
B ----- INVOICE -----							
C							
D Name.....				Invoice #.....			
E							
F Account #.....				Date.....			
G							
H	Part#	Description	Quantity	Unit Price	Extension		
I							
J		
K		
L		
M		
N							
V				TOTAL			
123456789012345678901234567890123456789012345678901234567890123456							
Screen Worksheet for Program Name - A:ARTOPRO							
Data Definitions:							
Field #	Name	Length	Type	Format			
1	Name	19	A				
2	Invoice #	8	A				
This is the Key Field							
3	Account #	8	A				
4	Date	10	A				
5	Part #	6	A				
6	Part Description	28	A				
7	Quantity	5	N	#####			
8	Unit Price	5	N	##.##			
9	Extension	8	N	#####			
Calculation Field (Stored)							
Calculation is - (F#(7))*F#(8)							
10		6	A				
11		28	A				
12		5	N	#####			
13		5	N	##.##			
14		8	N	#####			
Calculation Field (Stored)							
Calculation is - (F#(12))*F#(13)							
15		6	A				
16		28	A				
17		5	N	#####			
18		5	N	##.##			
19		8	N	#####			
Calculation Field (Stored)							
Calculation is - (F#(17))*F#(18)							
20		6	A				
21		28	A				
22		5	N	#####			
23		5	N	##.##			
24		8	N	#####			
Calculation Field (Stored)							
Calculation is - (F#(22))*F#(23)							
25	Total	9	N	#####			
Calculation Field (Display Only)							
Calculation is - F#(9)+F#(14)+F#(19)+F#(24)							

Sample summary of program specification

as the key, followed by the Enter key. *MINI-QUIKPRO* will ask if you are sure, and you must verify your choice before going on.

Next you must specify which fields are to hold only numbers. This feature will reduce data entry errors when you use the program by accepting only numbers for an item. If you want to restrict information in a field to numbers (and a decimal point), you should specify that field as a "numbers-only" field.

Type the number of the field that you wish to be "numbers-only," and you will be prompted to give the Format for that field. The Format shows where the decimal point, if any, should be displayed. A

**KEEP
LOOPING
THROUGH
THE
SPECS
UNTIL
ALL
DESIRED
FIELDS
ARE SET**

**ALL
ARITHMETIC
IS
PERFORMED
FROM LEFT
TO RIGHT**

Format of 7 means that all seven spaces in the field may hold only numbers. A Format of 4.2 means that the first four spaces may be numbers only, the fifth space must be a decimal point, and the sixth and seventh spaces are numbers only. Remember that the Format must equal the total length of that field. When you finish specifying a field as numbers-only, you are asked if you wish to specify any other fields as numbers-only. Keep looping through this specification process until all desired fields are set.

When you reply that there are no more numbers-only fields, *MINI-QUIKPRO* will then list all the fields that have been defined as numbers-only, and ask if there are any more. If there aren't, simply answer "N" to this prompt to continue.

Note that there are times when you might not want to specify "numbers-only" for a field, even though it will not contain letters. For example, zip codes are all numbers, but in parts of the Northeast, they begin with 0. To keep this leading zero from being dropped, do not specify "numbers-only" for this field. If you wish to list telephone numbers with punctuation (such as "(800)555-1212"), you must not specify "numbers-only".

CALCULATION FIELDS

The next prompt will ask if there are any Calculation fields. These are numbers-only fields that are the result of calculations based on other fields on the screen. If you answer "N" for no, the program will skip to the next part of the process. If you answer "Y," you will be asked for the field number identifying the Calculated field. After you enter the field number for a numbers-only field, you will be asked if this field is to be (S)to red or (D)isplayed Only. "Stored" Calculated fields have their values stored in the file with the rest of the data entered on a screen. "Displayed Only" fields are not stored in the file.

After you have specified whether the field is to be stored or not, you will be asked to enter the calculation. You may use the four standard mathematical operators: + for addition, - for subtraction, * for multiplication, and / for division. You specify the fields on a screen with the notation F#(n), where "n" is the number of the field on the screen. For example, the calculation

$$F\#(1) + F\#(2) * 10$$

would add the values in the first two fields, and multiply this sum by 10 (of course fields 1 and 2 themselves must be numbers-only fields for this calculation to be valid).

Note that you may not use parentheses to control the order of operations in a calculation. All arithmetic is performed from left to right, without regard to the operations involved. Should you need more complex formulas, you must modify the appropriate sections of the BASIC program which *MINI-QUIKPRO* generates.

After you have completed the Calculated field section, you will be asked to enter descriptions for each of the fields on the screen. You have up to 30 characters for each field, which you may use to describe the information contained in each field. These comments will become part of the program created by QuikPro I, making it easier for you to identify the different variables and sections of the program, but they will not appear anywhere when your program itself is run.

Next you will be asked to give a name for the Data File to be created and used by your program. File names must contain only letters, and may have up to eight letters in all. Note that if you specify which drive the file is to be placed on, this becomes part of the program, and your program will always expect to find that file on that drive. You may, of course, modify this part of the program once it has been generated by *MINI-QUIKPRO*.

You will next be asked for the name of the program that you want to create. This works in the same way as the Data File name above. Once this is complete, *MINI-QUIKPRO* will offer to print out a screen worksheet. This is a handy feature that gives you a printed copy of all the details that you gave to define the screen and fields. We urge you to use it.

In the final step, *MINI-QUIKPRO* asks if it is OK to begin generating the program. Reply:

OK 

and your program will be generated automatically. This may take some time, but *MINI-QUIKPRO* will keep you posted as it goes, using messages on the screen. When it is finished, you will be given the opportunity to run your new program or return to DOS.

RUNNING YOUR PROGRAM

Once *MINI-QUIKPRO* has generated your program, you will be able to list, copy, modify and run the program as you choose. Since it requires BASIC, it is best to make sure that you save the program on a disk with DOS and BASIC already on it. To run your program, place the disk with BASIC and your program in drive A, and at the DOS prompt, type:

BASIC progname /S:255 

(Instead of *progname*, use the program name that you specified when *MINI-QUIKPRO* created your program.) This command will load BASIC and your program, and it will then run automatically.

You will next see your data entry screen, with a prompt at the bottom, asking whether you wish to (A)dd a record, (G)et a record, (S)earch for a record, or (E)nd the program. You simply type the first letter of one of these four commands followed by the Enter key.

The first command you will use will be the (A)dd command to enter data. This will move the cursor to the first position in the first field, and wait for you to enter data. Enter the data for each field, and press the Enter key to advance to the next field. If you make a mistake in the current field you can use the Backspace key to delete the error. If your mistake is in a previous field, press the "<" key at the first position in a field followed by the Enter key, and you will move back to the previous field. When you have completed the last field on the screen, the program will write the information to your data file.

The (G)et command allows you to look up a record using its key field. After you type "G" and press the Enter key, the cursor will jump to the key field on your data entry form and wait for you to enter the key field value that you want to retrieve. If the program cannot find a record with that key, it will display a message saying so. If there is a record with that key field value, it will be displayed on the screen. You may then (C)hange or (D)elete the record, go on to the

 **YOU
WILL BE
ABLE TO
LIST,
COPY,
MODIFY,
AND RUN
THE
PROGRAM
ONCE
IT IS
GENERATED**

**YOU
MAY ALSO
SEARCH
FOR
RECORDS
BASED ON
VALUES
OTHER
THAN THE
KEY FIELD
VALUE**

(N)ext record with that key value, or return to the initial prompt. The first three are done by typing the letter of the command; return to the initial prompt by simply pressing the Enter key. If you elect to change the record, the cursor will go to the first field on your data entry form. To replace a value, just type in the new value. To leave the current field value unchanged, press:



You may also search for a record or records based on values other than the key field value. After selecting the (S)earch option, you will be asked what string you want to search for. The program will take the characters that you enter and search all fields of all records for a match. When a match is found, the record will be displayed on the screen and you will be prompted to either (S)top the search process, or tell the computer to find the (N)ext record containing the search string. When there are no more records with this string, or if you choose to stop the search, the program will return automatically to the primary action menu.

Search can be a handy feature for a type of selective record review, or if you can't remember the key value of a record you want to change. Search will only display records, but not let you change or delete them. Nevertheless by making a note of the key value displayed, you can come back to this record with the Get command, and then change or delete the record.

The last of the four main action commands is (E)nd. This exits the program and returns you to DOS.

After all the effort to create your program and then enter your data, you may feel frustrated by the rather severe limitations on what you can do with your information. Don't get mad, get *MINI-QUIKPRO* Part II, coming in the next issue of *PC Disk Magazine*. Part II will provide the same manner of streamlined, non-technical procedures for the creation of report programs. With Part II you will have the flexibility to easily extract and display all the information you have generated with *MINI-QUIKPRO* Part I.

BACK ISSUES OF PC DISK MAGAZINE

Add to your *PC Disk Magazine* collection today, while copies are still available. The more complete your library of back issues, the more enjoyment and use you can get from your personal computer. The earliest available back issue is Volume 1, Number 1. Order any issue you want, being sure to specify volume and issue number, and quantity, for each. This offer is subject to availability. If a particular issue is out of stock, your payment for that issue will be refunded promptly.

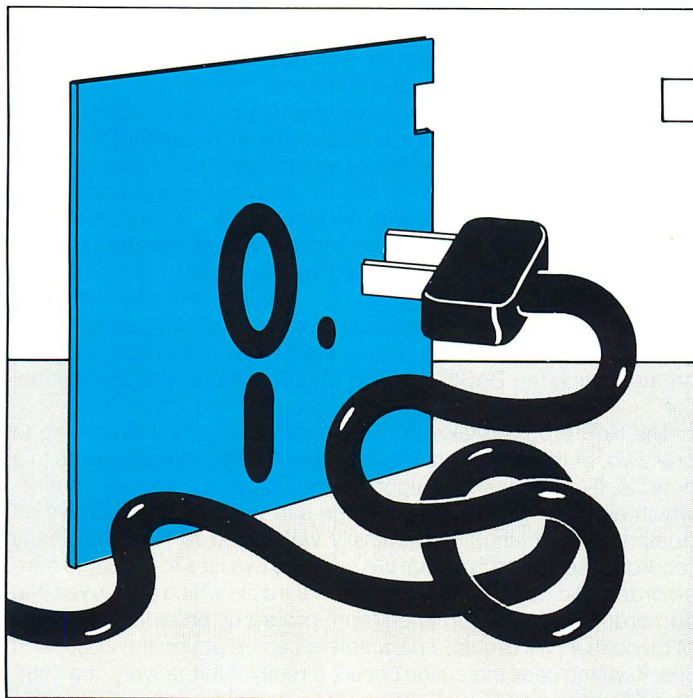
Back issues of *PC Disk Magazine* are priced at \$31.95 each, which includes postage and handling. Please write to:

PC Disk Magazine

CN 1916

Morristown, N.J. 07960

Be sure to include volume and issue number, as well as the number of copies desired, and the address to which you would like those copies sent. Please enclose your payment with your order.



SORT/MERGE II

By Louis J. Cutrona, Jr.

Special Requirements: None

Files Used: SORTF.EXE

MERGEF.EXE

It's one thing to write a program to maintain your club mailing list or recipe file, but when you try to make the files really useful, you discover that you always want the information sorted in a different way. Then you decide you want to combine two similar lists, but don't want to retype one to add it to the other.

In the last issue of PC Disk Magazine, we offered Sort/Merge I, a pair of programs that worked on ASCII files. Now we present SORT/MERGE II which handles any fixed-length file and supports all data types used by BASIC. You specify which file to sort, or which two files to merge, and the program steps you through questions and then writes the new file just the way you want it. You can even skip the questions by specifying all the needed parameters on your command line. Either way, the process is fast and easy.

BACKGROUND

SORT/MERGE II works with standard fixed-length files. These are most often files created by a program, containing multiple fields of differing data types (character, integer, etc.). The information in the file must be broken up into individual records, each one being the same length. This length is specified at the time the file is created, and you must know this file length in order to use *SORT/MERGE II*.

The Sort program and the Merge program recognize all four data types used in BASIC: character strings, integers, single precision numbers, and double precision numbers.

Character strings are sequences of single-byte characters. Character string variable names in BASIC end with the symbol "\$". Integer numbers require two bytes of storage and may represent any whole number from -32,767 to 32,767. They are created by the BASIC function MKI\$ and end with the symbol "%". Single precision numbers represent floating point decimal values, and require four bytes of storage. They are created by the BASIC function MKS\$ and end with the special symbol "!". Double precision numbers also represent floating point decimal values, but with a higher degree of accuracy, and thus require eight bytes of storage. They are created using the BASIC function MKD\$ and end with the symbol "#".

The Sort program takes a fixed-length file, sorts it according to one, two, or three different fields, and writes the sorted records to a new file. It uses a fast sort algorithm called the "Quicksort" routine, which employs a partition-exchange sort based on a recursive algorithm. This method is especially well-suited to files with many records. The Merge program will take any two files in the same sorted order, and combine them to form a third file which preserves that sort order. Both the Sort and Merge programs provide two means of invocation. All required parameters can be put on the command line, in which case execution occurs directly. Alternatively, the command line can be entered with just the relevant files specified. In this case a second parameter screen appears, whereon all the field particulars can be filled in. This assists the specification process. When you complete this screen, you can then start execution.

SORT

To use the Sort program, you must be in DOS. The file to be sorted can be on the same diskette as the Sort program (SORTF.EXE), or on a separate diskette. The same applies to the output file that will be created by the program. You should use the appropriate disk drive prefixes for your file specifications. Whatever your drive selections, begin by putting your *PC Disk Magazine* diskette in the default drive.

There are two ways to invoke the Sort program, one with the use of a second, field specification screen, the other with all parameters on a single command line. We will discuss the use of the parameter screen first, since it clarifies operating requirements and is easy to use.

With your *PC Disk Magazine* in the default drive, type:

SORTF Infile Outfile 

where "Infile" is the name of the file you wish to sort and "Outfile" is the name of the new, sorted file that you wish to create.

For example, say you have a list of addresses which you created in order of last names and the file is called NAMES.DAT. You now need this file's information in zip code order. You might call the new file ZIP.DAT. You could start the Sort program with the command:

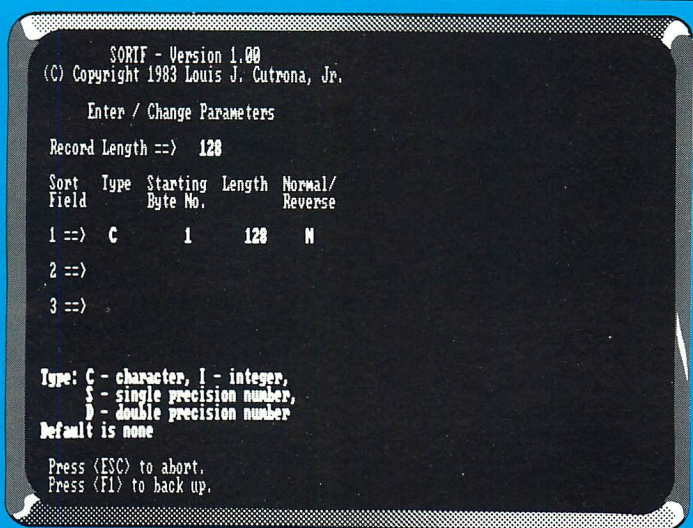
SORTF NAMES.DAT ZIP.DAT 

SORT PARAMETERS SCREEN

The Sort program next presents the Enter/Change Parameters Screen. Here you specify the different parameters which tell the program how to sort the file. After each parameter you enter, press:



to register your entry and proceed to the next field. You *must* specify the record length. You *may* specify up to three sort fields. For a given field you must specify the data type of the field, the starting byte number, the length, and the sort order (ascending/descending) for the field.



SORT prompt screen

The record length is the number of bytes contained in each fixed-length record of the file. This is determined by the program that created the file in the first place, and, as mentioned above, you must know this number in order to use the program.

The next three lines contain a set of four parameters each, by which to specify up to three sort fields. This means that if you are sorting a list by zip code, you can also have the information sorted so that all the names with the same zip code are in alphabetical order (the names would be a second sort field).

The sort field type parameter may be (C)haracter, (I)nteger, (S)ingle precision, or (D)ouble precision. If you do not specify the field type, the Sort program will assume the field is a character field.

**ALL
NAMES
WITH THE
SAME
ZIP CODE
CAN BE
SORTED
INTO
ALPHABETICAL
ORDER**

**THE SORT
CHECKS
YOUR
PARAMETERS
TO MAKE
CERTAIN
THAT
THEY ARE
ACCEPTABLE**

The starting byte number parameter is the relative location of the first byte of the sort field, counting the first byte in the fixed-length record as "1". Thus, if you want to sort by zip code, and the zip code field starts with the 56th character of each record in your file, then the starting byte number would be 56. If you do not specify a starting byte number, the Sort program will start the sort field with the first byte of the record (starting byte number 1).

The length parameter is the length of the sort field in bytes. If the field is a character field, then the length is the number of characters in the field, since each character takes one byte of storage. Integer numbers take two bytes apiece, so, for our zip code example, you would enter 10. Single precision numbers require four bytes, and double precision numbers require eight bytes. If you do not specify the length, the Sort program will assume that the length of the sort field extends from the starting byte number to the end of the record.

The direction of the sort may be (N)ormal or (R)everse. Normal sort order is ascending; "A" comes before "Z." Normal is the default order, so you do not need to enter "N" if this is your choice. An "R" in the fourth column will result in a reverse order sort, with "Z" coming before "A."

You may fill out the Parameter Screen for up to three sort fields. If you only wish to use one sort field (in other words, if you only wish to sort by one field), simply press the Enter key in the Type column for Sort Field #2. The program will then perform the sort, write the new output file, and return you to DOS. In a similar way, you may use just two sort fields by pressing the Enter key at the Type column for Sort Field #3.

While you are entering the information on the Parameter Screen, the Sort program checks your parameters to make certain that they are acceptable. The acceptable values for a field appear in a message area just below the sort fields. Of course the messages change as you move from field to field. For example, only the letter keys and the Enter key will work in the Type and Direction columns; press any other key and the program will beep and display a message explaining the error. Only the number keys, the Backspace key, and the Enter key will work in the other two columns.

Should you make a mistake and want to change an entry, press:

F1

This will move you back to the previous value so that you may make changes. You can abort the entire specification process at any time until the sort itself by pressing:

Esc

Once specification is complete and sorting begins, the program will stop and display an error message if there is not enough memory available to sort the input file, if there is not enough room to write the sorted output file, if the input file cannot be found, or if there is something incorrect about the sort field parameters.

DIRECT SORT

Once you gain confidence working with the Sort program, you may want to bypass the Parameter Screen entirely. This can be done

with the second method of sort invocation—specifying all the parameters on the command line. The format is:

SORTF Infile Outfile/parameter list/

The parameter list format is:

/Record Length/ sort-field1/ sort-field2/ sort-field3

As described above, "record length" is the length of the fixed-length records in the file to be sorted. You may just specify the record length without any parameters for the sort fields. In this case, the Sort program will treat each record as a single series of characters and will sort on the entire record so interpreted.

You may specify parameters for up to three sort fields (the same as on the Parameter Screen), using the following format for each field:

/type starting-byte-number length order/

Note that only spaces separate the different parameters in each field. The parameters of the different sort fields are separated by a slash (/). If you do not specify all the parameters for a sort field the Sort program will make the same assumptions as it does on the Parameter Screen.

MERGE

The Merge program takes two fixed-length record files that have the same size records and are in the same sorted order, and writes the combined data from the two as a third sorted file. This gives you one large file that contains all the information from the other two and that is sorted in the same order as the two original files.

In order for the Merge program to work properly, two important conditions must be met. Both of the files which will be combined must have the same record length and be sorted in the same way (by up to three sort fields).

As with the Sort program, you must be in DOS to use Merge. Likewise the two files to be merged, as well as the resulting file, can be referenced in any drives that your system supports with appropriate drive letter prefixes. Whatever your drive references, begin by putting your *PC Disk Magazine* diskette in the default drive.

Like Sort, Merge also supports two modes of invocation: parameter screen and command line only. For parameter screen invocation, start the program by typing:

MERGEF Infile1 + Infile2 Outfile 

where "Infile1" and "Infile2" are the files to be merged, and "Outfile" is the new, merged file to be created.

For example, let's say you have two lists of addresses that you have sorted in order of the last name, and the files are called NAMES1.DAT and NAMES2.DAT. If you now need to combine the information in these files into one large file, called BIGFILE.DAT, the command to start the Merge program could be:

MERGEF NAMES1.DAT + NAMES2.DAT

BIGFILE.DAT 

  **OTH
FILES
MUST
HAVE
THE SAME
RECORD
LENGTH
AND SORT
ORDER**

The Merge program next presents the Enter/Change Parameters Screen. Just as with the Sort program above, you will enter the different parameters which instruct the program on how you want the files merged. In order for the Merge program to combine the two files in proper order, you must specify the record length and the parameters that represent how the two files are sorted. This will require specifying the type of sort fields, the starting byte number, the length, and the direction of the sort for up to three sort fields. Each of these factors is described above in the Sort Parameter Screen section.

If the files that you wish to use are sorted on only one field, simply press:



in the Type column for Sort Field #2. The program will then perform the merge, write the new output file, and return you to DOS. In a similar way you may specify just two sort fields by pressing the Enter key at the Type column for Sort Field #3.

Just as in the Sort program above, the Merge program checks your parameters as you enter them, to make certain that they are acceptable, and as with Sort, you may correct a mistake by pressing:



to move the cursor back to the previous value.

The program will stop and display an error message if there is not enough room to write the merged output file, if either one or both of the input files cannot be found, or if there is something incorrect about the merge field parameters.

Once you gain confidence working with the Merge program, you can bypass the Parameter Screen entirely, just as you can with the Sort program. You can specify all the parameters at the time you give the command to run the program. The format is:

MERGEF Infile1 + Infile2 Outfile / parameter list /

The parameter list format is the same as for the Sort program:

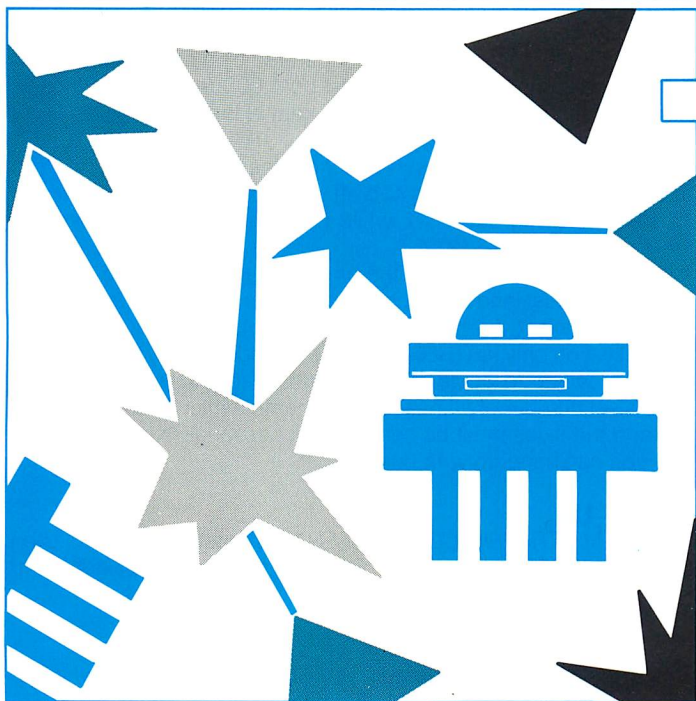
**/record length / merge-field1 / merge-field2 /
merge-field3**

You may specify parameters for up to three sort fields (the same as on the Parameter Screen), using the following format for each field:

/ type starting-byte-number length order /

If you do not specify all the parameters for a sort field, the Merge program will make the same assumptions as on the Parameter Screen.

After calling the program in this way, the program will perform the merge without further prompting, write the merged output file, and return you to DOS.



AIR TRAFFIC CONTROLLER

By Will Fastie and William Appelbaum
Original Design by David Mannering

Special Requirements: None
Files Used: ATC.EXE

In AIR TRAFFIC CONTROLLER, you must simultaneously guide up to 26 aircraft, flying with different speeds, altitudes, and directions, to their destinations, while avoiding mid-air collisions and fuel depletion.

You control the directions, turns, altitudes, and patterns of the aircraft as they enter and exit your airspace, take off and land. You must route these criss-crossing behemoths to their appointed destinations while keeping them out of harm's way. AIR TRAFFIC CONTROLLER isn't without risks—command a wrong turn or keep a plane in a holding pattern too long and you could send one of the silver birds plummeting to earth (thus terminating the game).

The game isn't heartless, though. By controlling the duration of your shift and the total number of aircraft in the game, you can regu-

late its complexity and difficulty. Everything is supplied; all you bring are the Roloids.

RULES OF THE ROAD

You win only after you have successfully routed all aircraft to their planned destinations. Failure to obey certain laws of the sky, or the laws of gravity, has disastrous results, suffered by both you and that pilot up there.

The first rule is to keep aircraft separated by a safe distance. Aircraft at the same grid location must be 1,000 feet apart vertically. Aircraft at the same altitude must be kept at least three miles apart at all times.

Second, the aircraft must reach their destination before they run out of fuel. Of the two types of craft you must direct, jets can fly for 15 minutes (program time), while propeller planes have enough fuel for 20 minutes.

Third, you must guide all the aircraft to their assigned destinations. They must land at the proper airport or leave your airspace at the proper point ("fix" in air controller lingo).

Finally, you only have a certain time period (which you can initially set) in which to guide all the aircraft to their destinations. You have a set number of planes for each game (which you can also alter) and the skies must be clear and all passengers deplaning before you can hang up your headset.

START UP

To start *AIR TRAFFIC CONTROLLER*, you must be in DOS. Then place your *PC Disk Magazine* diskette into your default drive and type:

ATC 

This starts the game. By default, you'll have 35 minutes in which to successfully route 26 aircraft to their destinations. However, beginners might want to try something less demanding and nerve-wracking. To alter the duration of your shift and the number of aircraft, start the program by typing:

ATC time aircraft 

where "time" is the length of your shift in minutes, from 16 to 99, and "aircraft" is the number of aircraft you will handle on your shift, from 1 to 26. The greater the ratio of time to aircraft, the simpler your shift will be. The game will not actually last 99 minutes, because you may advance the time in increments of 15 seconds by pressing the comma (",") key.

THE SCOPE OF THE GAME

Your screen is a computer-assisted radar scope patrolling 500 square miles. North is at the top of your scope. Displayed on the screen are 500 dots - 25 across and 20 deep. The two airports are represented by the symbols "#" and "%". There are also two multi-purpose navigational beacons ("navaids," in industry parlance) shown by asterisks ("*"). The numbers "0" through "9" along the edges of the scope are "fixes" where aircraft can enter or exit the airspace. The distance between any two adjacent dots, whether drawn horizontally, vertically, or diagonally, is one mile. On

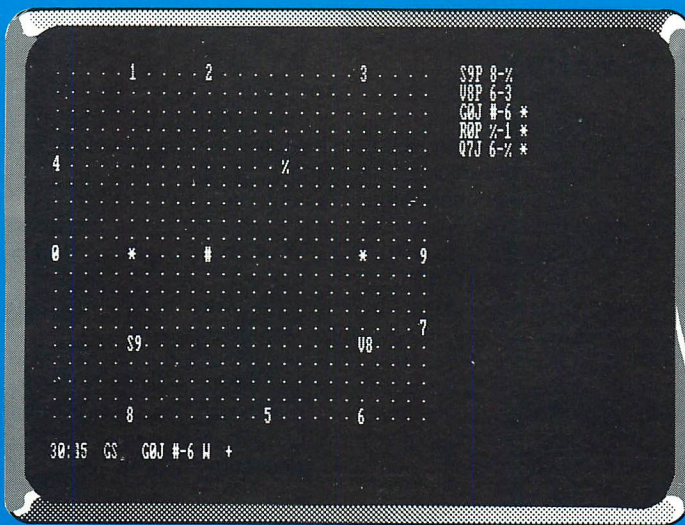
the lower left-hand corner, below the grid, is a digital readout of the time remaining in your shift. Immediately to the right is the communications line, which shows your instructions to pilots and their responses. Your scope display is not updated continuously, but rather at 15 second intervals. If you accelerate the clock manually by pressing the comma key, your scope will be updated with each keystroke.

FLIGHT PLANS

Aircraft in your sector will be doing one of three things:

- They will enter the airspace at one fix, pass through it, and exit at another fix. Their course need not be altered, but you must ensure that they exit at an altitude of 5,000 feet.
- They will enter the airspace and require guiding to one of the airports for a landing.
- They will start at an airport and need instructions in order to take off and fly to a specific exit fix (or to the other airport).

Each plane has a flight plan which will tell you which of these possibilities has been chosen for the aircraft.



ACH
PLANE
HAS
A FLIGHT
PLAN

The friendly skies of AIR TRAFFIC CONTROLLER

One minute before an aircraft becomes active (before it appears on your screen), its flight plan will be displayed in the upper right-hand corner of the screen, to the right of the grid. A flight plan looks like this:

K7J 0-% *

"K" is the aircraft call-sign. Since this can be any letter, you may have up to 26 planes. "7" indicates the aircraft's starting altitude, here 7,000 feet. Aircraft fly in airplanes 1,000 feet apart vertically. You may command a plane to fly at any level between 0 (ground level) and 5 (5,000 feet), although planes will enter your airspace at a higher altitude.

 **THE
PILOT
RESPONDS
WITH
"ROGER"
IF ABLE TO
PERFORM
THE
ACTION**

"J" designates the type of aircraft; "J" stands for "jet" and "P" for "propeller plane." Jets fly at 240 m.p.h. (one mile every 15 seconds) and prop planes fly at half this speed. Thus jets will move one dot on your scope with every update, while prop planes will move only every other update (except where you press the comma key to advance the time manually).

The next character tells the origin point of the craft. This may be an entry point (fix) on the grid (O through 9) or one of the two airports (# or %), meaning the craft is waiting to take off. The character after the hyphen is the destination. Again, this can be a fix, or, if the plane is routed to land, an airport. The final asterisk will appear if the plane has filed a flight plan but has not yet entered the airspace or has not yet been instructed to take off. When an aircraft completes its flight, its flight plan is removed from the display.

Active aircraft are indicated at their proper positions on the radar scope with a two-character symbol:

K2

where "K" is the aircraft call-sign and "2" is the current altitude (2,000 feet). The symbol is always visible unless another aircraft flies over at a higher altitude.

COMMANDS

Commands to aircraft pilots consist of two or three letters. The first letter is always the call-sign of the aircraft in question. The second letter is the command, and the third letter is a parameter which may be required by the particular command.

Commands are echoed in the command area of the screen to the right of the time readout and are acknowledged in the response area. The pilot responds with "ROGER" if able to perform the action, "UNABLE" if the action cannot be taken, and "SAY AGAIN?" if the command is not understood. If the aircraft is not yet active or has completed its flight, the response is dead air ("-----").

Commands are processed as soon as the final letter is typed (you need not press the Enter key to register your command). Before pressing the final letter, the command may be erased with the Enter key, the Delete (backspace) key, or the Space Bar.

The two main types of commands affect either aircraft altitude or direction.

Altitude commands look like this:

KA4

where "K" is the call-sign, "A" is the altitude, and "4" is the desired altitude. Aircraft climb and descend at the rate of 1,000 feet per mile. Regardless of their present altitudes, they respond only to altitude commands specifying 5,000 or below.

Aircraft may be commanded to turn either left or right, in increments of 45 degrees. In issuing a turn command you specify the extent of the turn with a multiplier from 0 to 4. The turn looks like this:

BL2

This command instructs aircraft B to make a 90-degree turn to the left. Left and right are always given from the point of view of the pilot, not the controller. Aircraft will continue on their present course for one mile before changing direction; they begin their turn at a rate of

one 45-degree increment per mile. A turn of 0 in either direction means that the aircraft maintains its present heading, just as it would if you used the command to proceed, "P."

The hold command "H" instructs the pilot to place the aircraft in a holding pattern (a continuous left-turning circle) when it passes directly over a navaid. The hold will continue until the plane receives a turn or approach command. The proceed command ("P") instructs a craft to continue flying in the direction it was headed at the time the command was issued.

TAKEOFF AND LANDING

The two airports have only one active runway each. Takeoffs and landings must be in the direction of these runways. To take off, you must instruct a plane to ascend to an altitude of 1,000 to 5,000 feet. The plane's marker will move away from the airport, automatically following the direction of the runway (northwest from airport "%" and west from airport "#"). You then must route it to its final destination.

Landings are accomplished by guiding the aircraft to altitude "0." The plane must approach the airport in the direction of the active runway: northwest for airport %; west for airport #. Once the A0 command has been issued, the plane no longer responds to your commands. Control is turned over to the airport air traffic controllers. If you have instructed the plane to land at the wrong airport, or if you give the landing command too late, the landing will abort. If the plane passes over the airport, it will be called off by the airport controllers and continue away from the airport at an altitude of 1,000 feet. If however, you miss the airport after an A0 command, that plane is now out of control altogether, and disaster is only a matter of time.

FLIGHT 54 WHERE ARE YOU?

Things get fast and furious in the tower while controlling six or more aircraft. It's quite easy to lose track of a plane's current position and heading. To remedy this, the air traffic controller can demand to know the status of any aircraft.

The status commands may be issued to any active aircraft. Its digital transponder answers with information regarding its current navigational and operational status. The message is displayed on the response line and looks like this:

W3P #-4 NW +

where "W" is the aircraft call-sign, "3" is the current altitude (3,000 feet), "P" is the aircraft type (prop), "#" is the origin (airport #), "4" is the destination (exit fix 4), "NW" is the current heading (direction of next movement), and "+" indicates the amount of fuel left, in minutes ("+" means 10 or more). If the aircraft which is called is not active or has completed its flight, the response is dead air.

INSTRUMENT FLYING

The two nav aids serve several functions. As already explained, they are the operating points for the hold commands. The west (left) nav aid automatically puts a holding pattern on any aircraft which flies over it bound for an airport. The east (right) nav aid can be used

**THE
AIRCRAFT
TO
ALTITUDE
"O"
TO LAND**

to make instrument approaches to either airport (note that it is located where the approach paths for the two airports intersect). If plane B is flying directly at this navaid, the command

B%

will instruct it to fly to the navaid, make a quick turn toward airport % and follow the approach path. Planes in a holding pattern can also be given the approach command. They will execute it when they circle over the navaid. Do not give an aircraft the approach command from the west navaid, or you will send the plane toward nonexistent airports. A collision will be noted in the command/response line and will end the game. The game is also ended if any plane runs out of fuel or tries to exit the airspace at a non-fix point, or at an altitude other than 5,000 feet.

To leave the game voluntarily and return to DOS, press:

Esc

This is the approved method. The other option is to trash a plane, which, while not the accepted method, is more fun.

COMMAND SUMMARY

Commands:

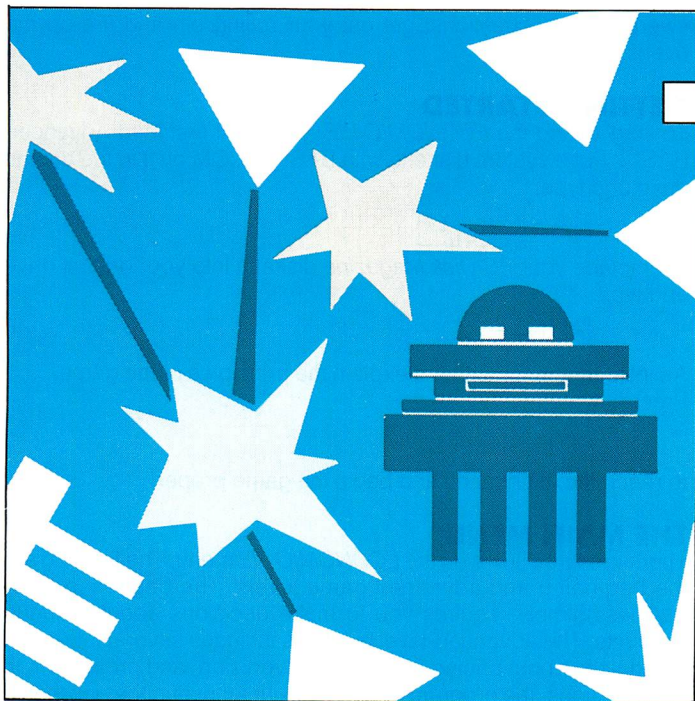
< aircraft > A0	cleared to land (no further commands accepted)
< aircraft > A < 1-5 >	go to assigned altitude, ascending 1000 feet/mile (if at 0, take off)
< aircraft > M	maintain current altitude
< aircraft > L < 0-4 >	left turn n*45 degrees
< aircraft > R < 0-4 >	right turn n*45 degrees (turns cancel H, #, % commands)
< aircraft > P	proceed on current heading (cancels H, R, L commands)
< aircraft > H	left circle at navaid (assume holding pattern).
< aircraft > #	turn west at navaid, cancel H
< aircraft > %	turn northwest at navaid, cancel H
< aircraft > S	status of aircraft, if active

Requirements:

destination #	land level at 0, headed west
destination %	land level at 0, headed northwest
destination fix	exit level at 5 thousand
separation	different altitude or 3 miles
fuel	jets 15 minutes, props 20 minutes

Other:

- Commas advance time by 15 seconds.
- Landing aircraft have an implicit hold (H) command for the left navaid.
- Commands are executed upon detection of last needed keystroke.
- Upper or lower case input accepted.
- The Escape key terminates the game.



ECONOMIC PIE

By John Springfield

Special Requirements: None

Files Used: ECOPIE.BAS

Are you tired of the nine-to-five drudge of bosses and time clocks? Have you always wanted to be self-employed? Here's a program that will test some of your basic money-making skills and indicate how you might do if you managed your own small company. ECONOMIC PIE sets you up as a bakery owner, forces you to make certain business decisions, and calculates the financial consequences of those decisions. Putting on an apron to bake the ECONOMIC PIE might just teach you how to get your slice in the real world.

BACKGROUND

ECONOMIC PIE introduces some basic principles by posing typical questions in business and marketing. Lowering the retail price of an

**ARE
ENOUGH
NEW SALES
CREATED
TO JUSTIFY
THEIR
COSTS?**

item will encourage customers to buy more, but can you increase volume enough to offset the lower margin? Do advertising and promotion create enough new sales to justify their costs? How much inventory will satisfy customer demand without wasting money through overstocking? And how much of your profits should you allocate for unexpected expenses? *ECONOMIC PIE* allows you to tackle these problems at either of two levels of difficulty. To start answering these questions, get out your rolling pin and fire up the ovens.

GETTING STARTED

To start slicing the *ECONOMIC PIE*, you must first load Advanced BASIC. Put a system diskette with BASICA.COM on it in the default drive and type:

BASICA 

Then place your *PC Disk Magazine* diskette into your default drive and type:

RUN"ECOPIE 

An introductory screen will explain the principles of the game. Press:



to reach the main menu and begin the game proper.

THE MAIN MENU

There are three items on the *ECONOMIC PIE* menu. The first two are the beginning and advanced game levels. The beginning-level business simply requires you to make decisions about ordering products. The advanced-level business includes several interrelated choices about pricing, inventory, advertising, and promotion. In both versions, the program uses both your input and the unpredictability of the market to generate sales results.

When you select choices "1" or "2" to play the game, you are asked to enter your name before the game actually begins. If you have already answered this question during the current playing session, your most recent reply is displayed. To leave it unchanged, just press:



To change it, or if no name appears, just type in the name you want followed by the Enter key.

The third choice on the main menu allows you to close the doors of your business (exiting the program) and return to Advanced BASICA. While playing either the Beginner or Advanced game, you can stop the game at any time by pressing:



When you do so, a prompt appears at the bottom of the screen asking if you really want to quit. If you do, type "Y" and you will be returned to the main menu, from which you can start a new game or exit the program. Responding with any other key will cause the game in progress to continue.

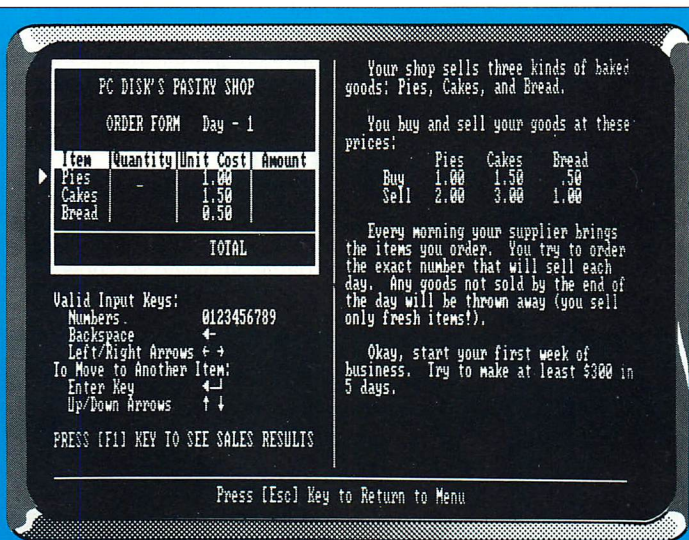
THE BEGINNER'S BAKERY

The first game is a simple exercise in ordering. The program's bakery sells three different items: pies, cakes, and bread. As owner, you order these items from a supplier at fixed costs and sell them at a fixed markup of 100%.

Pressing:

F1

starts the game, confronting you with your daily business decision: how much of each product to order. The program allows you to buy up to 99 units of each item daily, but you must order exactly as many of each as you think the public wants to buy. Ordering too many wastes money; ordering too few discourages customers and diminishes profits.



The game screen for the bakery business

Like anyone starting a new business, you'll have to guess at first, but experience will soon give you a feel for the market.

Enter your choices for the first day on the order form on the left side of the screen, using the valid input keys shown on the order form. You may also use the numeric keypad to enter numbers, but it is better to keep it switched to cursor control mode (NUM LOCK off) and to use the top row of the main keyboard for inputting numbers.

As you enter your order, the program will calculate the total cost for each item. You can change your order at any time until you press the F1 key. Once you do so, the program will process your order, deliver the pies, cakes, and bread, and sell them for you. The results of your first day's sales will then appear on the right-hand side of the screen.

On the top of the screen is a chart showing how many of each item you ordered, how many were sold, and how many were left over at the end of the day. In the middle is your daily cash flow,

**THE
PROGRAM
WILL
PROCESS
YOUR
ORDER,
DELIVER
THE
GOODS,
THEN
SELL THEM
FOR YOU**

After five days the program will tell you how well you've done for the week. The computer sets up a goal of \$300 profit per week, but don't expect to make that much when you start in business. The program gives you the option of trying again. At the end of each week, press:

Y

in response to the computer's "Try again?" question to restart the beginner's game. Any other key takes you back to the main menu.

Once you are comfortable ordering baked goods and are making a good weekly profit, you're ready for more complicated business decisions.

THE PROFESSIONAL'S PIE SHOP

The advanced level of *ECONOMIC PIE* places you in your own "Pie Palace." This store is run on a month-by-month basis for a full fiscal year. You have fixed expenses and variable expenses each month, and a stroke of the F1 key starts the game, showing you how these conditions stand when you take over at the end of the first month.

**THE
CURRENT
BUSINESS
DECISIONS
ARE
DISPLAYED
ON THE
LEFT SIDE
OF THE
SCREEN**

PC DISK'S PIE PALACE	
Business Decisions For Month 2	
Note: Cost of Pies now	\$1.00
Cost of Ads now	\$50.00
Number of pies ordered	500
Your selling price per pie	5.00
Number of newspaper ads	1
Promotional costs	100

Results For Month 1	
Income	
Pies Sold: 500 at \$5.00	2500.00
	=====
Expenses	
Fixed Costs	1700.00
Variable Costs	
Cost of Pies: 500 at \$1.00	500.00
Newspaper Ads: 1 at \$50.00	50.00
Promotional Costs:	100.00
Unexpected Costs:	10.00
Total Expenses:	2360.00
	=====
Profit:	140.00
Unexpected Cost:	
	=====
	Petty cash expense
	=====

Valid Input Keys:

Numbers 0123456789

Backspace ←

Left/Right Arrows ← →

To Move to Another Item:

Enter Key ↵

Up/Down Arrows ↑ ↓

Press [F1] Key to See Sales Results

Press [F2] Key to See Totals to Date

Press [Esc] Key to Return to Menu

The worksheet for this month is on the left; the results from last month are on the right.

On the right of the screen is the balance sheet for the first month. The shop sold 500 pies at \$5.00 apiece for an income of \$2,500.00. The expenses totaled \$2360.00, so you made \$140.00. The current business decisions are displayed on the left side of the screen. You

must decide how many pies to order, how much to charge for each pie, how many newspaper ads to place, and how much to spend on promotion (T-shirts, contests, and so forth). The wholesale cost of the pies and the cost of newspaper ads are also listed on the left side of the screen. If you agree with the current choices, press:

F1

and they will be implemented. The program will order the pies, sell them, place the ads, handle the promotion, pay the bills, and balance the books. The results for the second month will now appear on the right side and the decision sheet for the third month will be on the left.

The program will allow you to order as many as 999 pies, sell them for up to \$9.99 apiece, place as many as 9 ads, and spend up to \$999 on promotion. The valid input keys to make these entries are again displayed in the lower left section of the screen. Note that you must use a decimal point when entering your selling price.

Here is the real test of your business sense. Advertising and promotion will boost pie sales, as will lowering your selling price. But these steps also reduce your profits. Ordering the right number of pies will help you cut your expenses, but watch out for increases in the cost of pies and ads, as well as unexpected costs. Adjust all the figures until you reach the best combination of decisions, then implement them for the month by pressing the F1 key.

Any unexpected costs will be explained at the bottom of the right side of the screen. Study the results and use them to make your next set of monthly decisions. Pressing:

F2

at any time will show you your year-to-date totals. Your monthly income, expenses, and profits will be shown on the left-hand side of the screen.

When you have finished a year of business, the program will display your totals for the year and give you a chance to try again. Don't be discouraged. Very few businesses make a profit in their first year of operation.

Any time you wish to close up shop, press:

Esc

and you will be returned to the main menu.

Good luck, and bon appetit!

**ANY
UNEXPECTED
COSTS
WILL BE
EXPLAINED
AT THE
BOTTOM
OF THE
RIGHT SIDE
OF THE
SCREEN**

SUBSCRIBE NOW

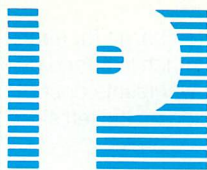
and save \$60 on a 6-issue subscription to *PC Disk Magazine*. Give yourself or a friend up to 72 ready-to-use programs for the IBM PC—a library of quality software—all for only \$119.

Write to:

PC Disk Magazine
P.O. Box 2462
Boulder, CO 80322

or call:

1-800-526-0666



PROBLEM RECOVERY

There really is not a great deal to say about problem recovery with *PC Disk Magazine*. If you use this software on the right equipment (IBM PC or PC-XT) running the appropriate system software, (DOS 1.1 or 2.0, Microsoft Advanced BASIC 1.1 or 2.0) you should experience no problems. Nevertheless, a few comments may resolve some more obvious difficulties.

Any BASIC program can be interrupted at any time by pressing:



If you do not see the OK message immediately, indicating that you are back in BASIC, press these keys again. This is a rather drastic but effective way of regaining control of the computer. You won't damage any of the programs in this way, since they're still intact on the diskette. However, you will lose any data you entered while the program was running.

If you interrupt a BASIC program you may find that the function keys no longer perform as they had before starting the program. This is because many *PC Disk Magazine* programs reset the function keys during execution, then restore the original settings upon completion. An interrupt causes an abnormal termination of a program, so the function keys are not restored. To correct this situation, simply exit from BASIC and then return to BASIC.

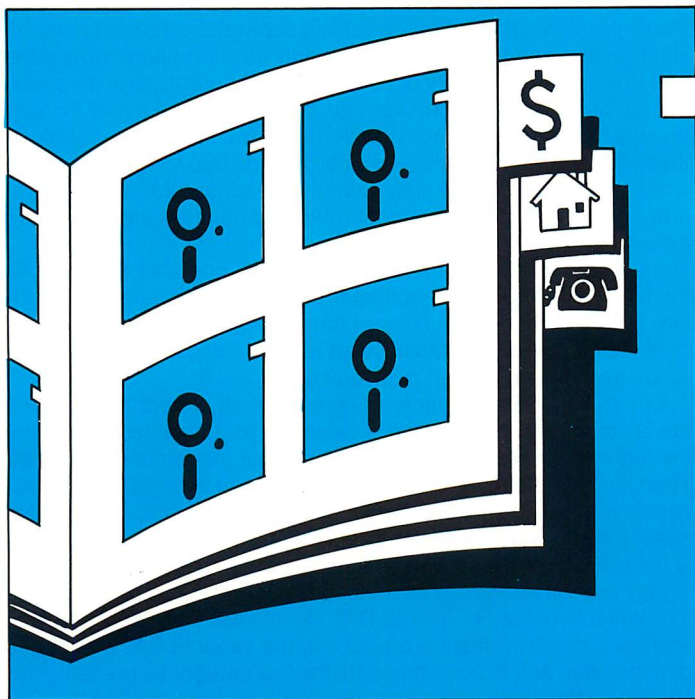
You may find at times that the cursor control keys are not working as they should. This is because the keys are not in cursor control mode. The key that switches these keys between numeric mode and cursor control mode is the Num Lock key. So to restore the keys to cursor control mode press:



If you try to send something to the printer when there is no printer, or when the printer is off or offline, you will hang the system. The computer will just sit there and will not respond to any keys pressed. After a few seconds you may get a BASIC error message indicating that the device was unavailable. The program that was running has been aborted, and you will be left in BASIC. If the computer does not put out any message, but just remains hung, you will have to say goodbye to whatever you were doing and re-boot your system.

Though we hope you will never need it, if you should find a "bug" in a *PC Disk Magazine* program, the address to write to is:

PC Disk Magazine
Dept. 741
One Park Avenue
New York, N.Y. 10016



1983 FEDERAL TAX RATES

By Morris Effron

Special Requirements: None

Files Used: FEDTAX83.DAT

FEDTAX83.BAS

They say only two things in life are certain: death and taxes. If that is so, then here's a reference file and associated program that can't help but be useful to you. You may not rush to use it upon receipt of your new issue, or be happy about what it has to say, but its utility is in a certain sense unavoidable.

The IRS actually publishes two types of tax tables: a preliminary set of tables, known as the "XYZ" tables, which provide very accurate estimates ($\pm 5\%$) of federal tax liability, and the complete 1040 tables, which provide an exhaustive, detailed list of all taxable amounts. The latter tables are used to complete your 1040 tax return, but they do not become available until after December 31 of

the relevant tax year. Therefore we have used the XYZ tables here to provide you with a very good estimate, but only an estimate, of your 1983 federal tax liability.

BACKGROUND

The data for the XYZ tables follows a simple structure. First, there are five tax categories when it comes to federal tax calculations: single persons, married persons filing separately, married persons filing jointly, heads of households, and the non-personal estates and trusts. Each category of course has a different set of tax rates, but all follow the same tabular structure. There is a list of thirteen taxable income levels, each of which has a base tax amount associated with it. Additionally there is a percentage rate for each tax level, which applies to the taxable income in excess of the base level amount, up to the next tax level (at which point the new base tax amount applies).

These five tables of information are present, one after the other, in the file FEDTAX83.DAT. Each table begins with its category heading, followed by thirteen lines of three numbers each (taxable income level, base tax amount, and rate on excess). This is an all ASCII text file, so you can print, edit or display it.

Computing estimated federal taxes is simple. Select the appropriate tax category, then find the tax level for your taxable income (i.e. the highest taxable income level which your income still exceeds). Then just add the base tax amount at that level to the difference between your taxable income and the income level amount (the "excess"), multiplied by the associated rate on excess. This is precisely the calculation performed by the program FEDTAX83.BAS.

This federal tax liability estimating program begins by reading in the contents of the FEDTAX83.DAT file into two variables. The federal tax category names are read into the variable CAT\$, which is dimensioned at 5 to accommodate the five categories present. The five tables themselves are read into the single variable TAXRATES. This single-precision variable has three dimensions. The first dimension identifies the tax category reference. The second identifies each of the thirteen tax levels within a category. The third dimension references the taxable income, the tax amount, and the rate on excess respectively for a given tax level within a given tax category.

To perform the necessary tax calculation, the program uses your choice of tax category, contained in the variable ICAT, as the first subscript for TAXRATES. With this subscript set, the appropriate tax table is searched to find the highest tax level for the amount you have entered, stored in the variable TAMT. When that level is found, and the related subscript stored in the variable ITAX, the tax calculation is performed using the taxable income amount, tax amount, and rate on excess for that level. The result is stored in the variable TAXAMT.

Before you fire your accountant, bear in mind that coming up with the correct amount for your taxable income is what the tax return ballgame is all about. This program will not help you do that. Rather it takes your taxable income figure as its starting point. Please also bear in mind that the figure you submit to this program should be taxable income after exemptions.

USE

To start the 1983 federal tax estimating program, first load Advanced BASIC into your PC. Put a diskette with the file BASICA.COM in your default drive and type:

BASICA 

Then put your *PC Disk Magazine* diskette in the default drive and type:

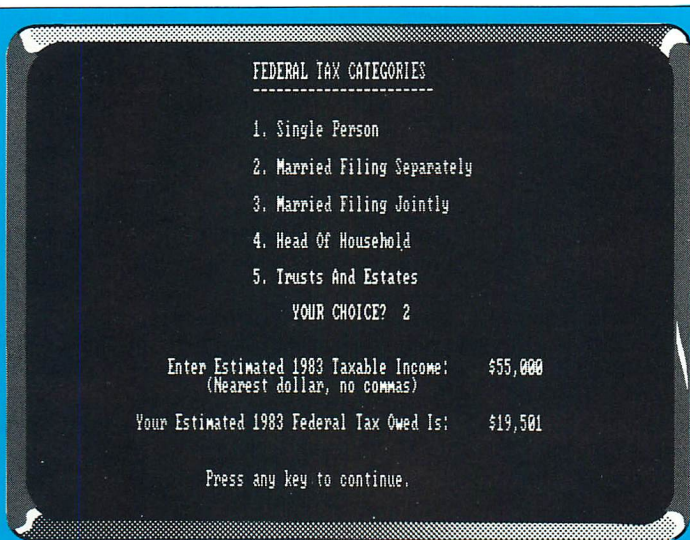
RUN"FEDTAX83 

You will first see a screen telling you that the tax table data file is being read. Then the tax calculation screen will appear.

Enter your choice of tax category by typing a number from 1 to 5 followed by the Enter key. Type:

0 

to exit the program and return to Advanced BASIC.



```
FEDERAL TAX CATEGORIES
-----
1. Single Person
2. Married Filing Separately
3. Married Filing Jointly
4. Head Of Household
5. Trusts And Estates

YOUR CHOICE? 2

Enter Estimated 1983 Taxable Income:  $55,000
(Nearest dollar, no commas)

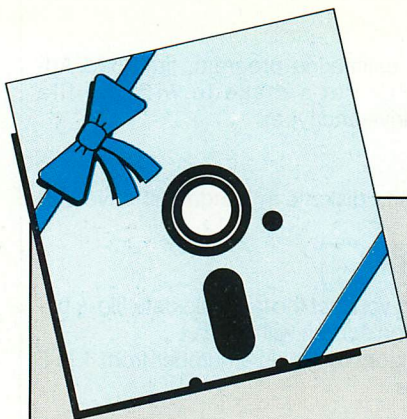
Your Estimated 1983 Federal Tax Owed Is:  $19,501

Press any key to continue.
```


COMPUTE
YOUR
1983
FEDERAL
TAXES
WITHIN
5%

Sample tax calculation—better run for shelter

With a category chosen you are then asked to enter your estimated 1983 taxable income. Enter any whole dollar amount, up to \$10 million, without commas or decimal points. The program will reformat your entry for visual consistency, then compute and display the estimated tax owed, based on the calculation described in the background section above. At this point you may press any key, as indicated by the message on the bottom of your screen, to return to the tax category choice. That's it!



EXPAND YOUR SOFTWARE LIBRARY

 This issue of *PC Disk Magazine* has already increased your software library for your IBM personal computer. If you haven't subscribed to *PC Disk Magazine*, just think of how easy it will be to expand your library with future issues delivered to your door at substantial savings. And if you already are a subscriber, wouldn't you like to share your knowledge with a friend and get them a gift subscription to *PC Disk Magazine*.

Either way, you can save \$60 with this introductory offer and give yourself or a friend up to 60 programs for as little as \$2.00 each! *PC Disk Magazine* is the solution to the problem of building and maintaining an up-to-date library of tested software programs.

You will get up to 10 varied and ready-to-use programs and files in every issue—complete with an illustrated user manual—for the ultimate in ease, efficiency and economy.

Every program in *PC Disk Magazine* is developed by experts, tested by the editors of *PC Disk Magazine* and published by the people who bring you PC, guaranteeing quality and satisfaction.

Use the enclosed card to subscribe today or to give a friend a gift. If the card is missing, please write to:
PC Disk Magazine, PO Box 2462, Boulder, Co. 80322
For even faster service, call our toll-free number below and use your American Express, MasterCard or Visa . . . today!

**CALL
1-800-526-0666
for FAST service
and get
6 issues of
PC Disk Magazine
for only \$119
a savings of 34%!**



MINI-QUIKPRO (PART II OF II)

The other half of our automatic program writer. Having used Part I to create your data maintenance program, use this part to automatically create report programs for that information.

INFLATION-PROOFER

Whether you're an economic optimist or pessimist, protecting your investments against inflation is an inevitable concern. Compare your strategies to the inflation-proofing theories of this program.

EARLY READER

A reading and spelling primer built on clear, vibrant displays.

BIORHYTHM

Check your physical, emotional and intellectual energy cycles for any timeframe. Display, print or file the results.

SETKEY

Use this simple utility to program your function keys with up to 40 different command strings of up to 80 characters each.

FILE MANAGER (PART I OF III)

The first module of this comprehensive file manager provides an integrated environment for executing DOS file commands.

BUG HUNT

An action game based on the delights and dangers of life among the lily pads.

JIGJAGS

We proudly introduce a new era in crossword puzzles that is also our first recurring feature. JigJags challenges you to construct a valid crossword puzzle from puzzle parts.

OENOPHILIAC DOSSIER—CHAPTER I

The first in a series of wine rating reference files.

CALLING ALL PROGRAMMERS!!

If you have written software for the IBM PC that you feel would be of use or interest to others, *PC Disk Magazine* would like to know about it. Publishing is no longer just for poets and prosaists, but now it's for programmers too. Ask for our Software Submission Plan by writing to:

PC Disk Magazine
Dept. 732
One Park Avenue
New York, N.Y. 10016

1. The first part of the report is a general statement of the purpose and scope of the study. It is followed by a brief review of the literature on the subject. The third part of the report is a description of the methods used in the study. The fourth part is a presentation of the results of the study. The fifth part is a discussion of the results and their implications. The sixth part is a conclusion. The seventh part is a list of references. The eighth part is an appendix. The ninth part is a glossary. The tenth part is an index.

CHAPTER I

THE PURPOSE AND SCOPE OF THE STUDY

The purpose of this study is to determine the effect of the use of the word "the" on the readability of a text. The scope of the study is limited to the use of the word "the" in the first sentence of a paragraph.

REVIEW OF LITERATURE

A review of the literature on the subject of readability has been conducted. The results of this review are presented in the following table:

TABLE I

Summary of the results of the review of the literature on the subject of readability.

DISCUSSION

The results of the study indicate that the use of the word "the" in the first sentence of a paragraph has a significant effect on the readability of the text.

CONCLUSION

The results of the study indicate that the use of the word "the" in the first sentence of a paragraph has a significant effect on the readability of the text.

REFERENCES

A list of the references used in the study is presented in the following table:

APPENDIX

A list of the appendices used in the study is presented in the following table:

GLOSSARY

A list of the terms used in the study is presented in the following table:

INDEX

A list of the subjects covered in the study is presented in the following table:

THE END OF THE STUDY

The study has been completed and the results are presented in the following table:

The study has been completed and the results are presented in the following table: